Front End

The Comment Anywhere front end is comprised of classes written in TypeScript/JavaScript. It is responsible for displaying the data to the user, capturing user interaction, converting those actions into Client-Server communication entities, and dispatching them to the server. It runs in the browser engine.

Cafe

**Class Description**: Cafe stands for "Comment Anywhere Front End". Cafe is the base class that is composed of other major classes used in the Front end. It is responsible for updating the State and listening for user input events on the DOM, and transmitting them to the fetcher when appropriate.

**Class Data Members**: Fetcher fetcher, State state, CafeNavbar navbar, Settings settings, Dispatcher dispatcher, string currentUrl

**Class Methods:** userChange(Server.UserProfile | undefined), pageChange(string), changeState(string), checkForResponses(), stateChangeEventReceived(), clientEventReceived(), settingEventReceived(Settings)

Fetcher

**Class Description**: Fetcher is responsible for dispatching requests to the server at the appropriate API endpoints and populating its responses object with the server responses.

**Class Data Members:** responses ServerResponse[]

**Class Methods:** fetch(string, client-server Communication Entity), getAndClearResponses(

State

**Class Description:** State holds the current state of the front end, including who is logged in and what window is being viewed. Cafe passes State to NavBar to realize a state change.

**Class Data Members:** string viewing, UserProfile ownProfile

**Class Methods:** loadProfile(UserProfile | undefined), setViewingTo(string)

CafeNavbar

**Class Description:** CafeNavBar displays navigation buttons for the user to move between states, holds the active window, and holds a general message display object.

**Class Data Members:** CafeCommentsWindow commentsWindow, CafeSettingsWindow settingsWindow, CafeModerationWindow modWindow, CafeAdminWindow adminWindow, CafeLoginWindow loginWindow, CafeRegisterWindow registerWindow, CafePWResetCodeWindow pwResetWindow, CafeNewPasswordWindow newPwWindow, CafeMessageDisplay globalMessage, CafeUserDisplay userHoverDisplay, CafeWindow currentlyViewing, HTMLButtonElement commentsButton, HTMLButtonElement settingsButton, HTMLButtonElement modButton, HTMLButtonElement adminButton, HTMLButtonElement loginButton, HTMLButtonElement registerButton, HTMLButtonElement logoutButton

**Class Methods:** commentsButtonClicked(), settingsButtonClicked(), modButtonClicked(), adminButtonClicked(), loginButtonClicked(), registerButtonClicked(), logoutButtonClicked(), setFromState(state:State), displayMessage(data:Server.Message)

CafeSettings

**Class Description:** CafeSettings holds comment sorting settings configured by the user.

**Class Data Members:** Boolean viewHidden, string sortBy, boolean sortAscending

Dispatcher

**Class Description:** Dispatcher is responsible for parsing an array of server responses and dispatching them to the appropriate objects around the front end for rendering to the user.

**Class Methods:** dispatch(serverResponse[], Cafe), dispatchMessage(Message, CafeNavbar), dispatchUserUpdate(UserProfile, Cafe), dispatchFeedbackUpdate(FeedbackRecord, FeedbackReportSection) , dispatchCommentUpdate(Comment, CafeCommentsWindow) , dispatchComments(Comment[], CafeCommentsWindow), dispatchUserProfile(UserProfile, CafeUserDisplay), dispatchLoginResponse(LoginResponse, Cafe), dispatchLogoutResponse(LogoutResponse, Cafe), dispatchPwResetCodeResponse(any, Cafe), dispatchPwResetReqResponse(any, Cafe), dispatchReqVerificationResponse(any, CafeMessageDisplay), dispatchNewPassResponse(any, CafeMessageDisplay) , dispatchVerifyResponse(any, CafeMessageDisplay), dispatchBanRecords(BanRecord[], BanRecordsSection) , dispatchCommentReports(CommentReport[], CommentReportsSection) , dispatchCommentReportUpdate(CommentReport, CommentReportsSection) , dispatchFeedbacks(FeedbackRecord[], FeedbackReportSection) , dispatchLogs(AdminAccessLog[], LogsSection) , dispatchModRecords(ModerationRecord[], ModActionsReportSection) , dispatchGlobalMods(GlobalModeratorRecord[], ModeratorsReportSection) , dispatchDomainMods(DomainModeratorRecord[], ModeratorsReportSection) , dispatchDomainReport(AdminDomainReport, CafeAdminWindow) , dispatchUsersReport(AdminUsersReport, CafeAdminWindow)

**CafeWindow**

**Class Description**: CafeWindow is the base class for all CafeWindows. CafeWindows hold all the content of CommentAnywhere, except for the navbar buttons and message. They correspond to the front-end states. Only one CafeWindow is visible at any given time, and it is displayed and hidden by the CafeNavBar instance.

**Class Data Members**: HTMLDivElement el

**Class Methods**: show(), hide()

**CafeCommentsWindow**

**Extends:** CafeWindow

**Class Description**: CafeCommentsWindow displays all comments for the current page. It is responsible for repopulating with new comments when comments for a new page are retrieved and updating comments as new ones are added and voted on.

**Class Data Members**: Comment[] data, lastSettings?: CafeSettings, Map<number, CafeComment> displayedComments,

CafeCommentSortDisplay commentSortSettings

Class Methods:populateNewComments(Comment[]), updateComment(Comment), resortComments(), updateFromSettings(CafeSettings)

CafeSettingsWindow

**Extends:** CafeWindow

**Class Description**: CafeSettingsWindow displays user settings and provides options for the user to reset their password or verify their email.

**Class Data Members**: CafeCommentSortDisplay commentSortSettings, HTMLButtonElement passwordResetButton, HTMLButtonElement requestEmailValidationButton, HTMLButtonElement verifyEmailButton, HTMLInputElement verifyCodeInput, HTMLInputElement verifyCodeSubmit

Class Methods:updateFromSettings(CafeSettings), verifyEmailClicked(), verifyCodeSubmitClicked(), passwordResetClicked()

**CafeModerationWindow**

**Extends:** CafeWindow

Class Description: CafeModerationWindow allows moderator interactions with Comment Anywhere. It is shown when a user clicks the moderation button on the Navbar, transitioning them to that state.

Class Data Members: ModeratorsReportSection moderators, ModActionsReportSection modActions, BanRecordsSection banRecords, CommentReportsSection reports

Class Methods:

**CafeAdminWindow**

**Extends:** CafeWindow

Class Description: CafeModerationWindow allows admin interactions with Comment Anywhere. It is shown when a user clicks the admin button on the Navbar, transitioning them to that state.

Class Data Members: FeedbackReportSection feedbackReport, LogsSection logs, domainReports: CafeDomainReportDisplay[]

Class Methods:

CafeUsersReportDisplay userReport, HTMLInputElement domainInput, HTMLButtonElement domainRequestButton, HTMLButtonElement usersReportButton,domainRequestButtonClicked(), showDomainReport(AdminDomainReport), usersReportRequestButtonClicked(), showUsersReport(AdminUsersReport),

**CafeLoginWindow**

**Extends:** CafeWindow

Class Description: CafeLoginWindow is displayed when the user clicks the "Login" button in the navbar and transitions to the login state.

Class Data Members: HTMLInputElement username, HTMLInputElement password, HTMLButtonElement submitLoginButton, HTMLButtonElement forgotPasswordButton,

Class Methods:submitLoginButtonClicked(), forgotPasswordButtonClicked()

**CafeRegisterWindow**

**Extends:** CafeWindow

Class Description: CafeRegisterWindow is displayed when the user clicks the "Register" button in the navbar and transitions to the register state.

Class Data Members: HTMLInputElement username, HTMLInputElement password, HTMLInputElement retypePassword, HTMLInputElement email, HTMLInputElement agreedToTerms, HTMLButtonElement submitRegister,

Class Methods:submitRegisterClicked()

**CafePwResetRequestWindow**

**Extends:** CafeWindow

Class Description: CafePWResetRequestWindow is displayed when the user transitions to the request password reset form state.

Class Data Members: HTMLInputElement email, HTMLButtonElement submitPWResetRequestButton,

Class Methods:submitButtonClicked()

**CafePWResetCodeWindow**

**Extends:** CafeWindow

Class Description: CafePWResetCodeWindow is displayed when the user transitions to the input password reset code state.

Class Data Members: HTMLInputElement code, HTMLButtonElement submitCodeButton,

Class Methods:submitButtonClicked()

**CafeNewPasswordWindow**

**Extends:** CafeWindow

Class Description: CafeNewPasswordWindow is displayed when the user transitions to the input new password state.

Class Data Members: HTMLInputElement password, HTMLInputElement newPassword, HTMLButtonElement submitButton,

Class Methods:submitButtonClicked()

**CafeWindowSection**

Class Description: CafeWindowSection represents an area of a window that performs some particular task or shows some collection of a data. CafeWindows may be composed of several CafeWindowSections.

Class Data Members: HTMLElement el,

Class Methods:

**FeedbackReportSection**

**Extends:** CafeWindowSection

Class Description: FeedbackReportSection is part of the CafeAdminWindow and displays information about user-submitted feedback.

Class Data Members: Map<number, CafeFeedbackDisplay> feedbacks,

Class Methods:

HTMLInputElement from, HTMLInputElement to, HTMLSelectElement feedbackType, HTMLButtonElement requestFeedbackButton,updateFeedback(FeedbackRecord), requestFeedbackClicked(), populateFeedback(FeedbackRecord[]), clearFeedback(),

**ModeratorsReportSection**

**Extends:** CafeWindowSection

Class Description: ModeratorsReportSection is part of the CafeModerationWindow and allows viewing of moderator assignment records.

Class Data Members: HTMLInputElement forDomainInput, HTMLButtonElement requestDomainModeratorsButton, HTMLButtonElement requestGlobalModeratorsButton,

Class Methods: Map<number, CafeDomainModDisplay> domainModRecords,

Map<number, CafeGlobalModDisplay> globalModRecords,

HTMLButtonElement clearModRecordsButton,requestDomainModsClicked(), requestGlobalModsClicked(), populateDomainModerators(DomainModeratorRecord[]), populateGlobalModerators(GlobalModeratorRecord[]), clearModsClicked(),

**ModActionsReportSection**

**Extends:** CafeWindowSection

Class Description: ModActionsReportSection is part of CafeModerationWindow and provides functionality for viewing moderation actions taken on comments.

Class Data Members: Map<number, CafeModRecordDisplay> modRecords, HTMLButtonElement requestModRecordsButton, HTMLButtonElement clearModActionsButton

Class Methods:requestModRecordsButtonClicked(), populateModActions(ModerationRecord[]), updateModAction(ModerationRecord), clearModActionsClicked()

**BanRecordsSection**

**Extends:** CafeWindowSection

Class Description: BanRecordsSection is part of CafeModerationWindow. It is used to display records of bannings to other moderators.

Class Data Members: banRecords: Map<number, CafeBanRecord>, HTMLButtonElement requestGlobalBansButton, HTMLInputElement forDomain, HTMLButtonElement requestDomainBansButton

Class Methods:populateBanRecords(BanRecord[]), requestGlobalBansClicked(), requestDomainBansClicked(), clearBanRecords()

**CommentReportsSection**

**Extends:** CafeWindowSection

Class Description: CommentReportsSection is part of CafeModerationWindow. It is responsible for displaying comment reports to moderators.

Class Data Members: Map<number, CafeCommentReportDisplay> reports, HTMLButtonElement viewReportsButton,

Class Methods:populateCommentReports(CommentReport[]), viewReportsClicked(), updateCommentReport(CommentReport), clearCommentReports()

**LogsSection**

**Extends:** CafeWindowSection

Class Description: LogsSection is part of CafeAdminWindow. It is responsible for populating log entries.

Class Data Members: HTMLInputElement forUser, HTMLInputElement forIP, HTMLInputElement forDomain, HTMLInputElement startingAt, HTMLInputElement endingAt, HTMLButtonElement submitLogsRequestButton, Map<number, CafeLogDisplay> logs

Class Methods:submitLogsRequestClicked(), populateLogs(AdminAccessLog[]), clearLogs()

**UIInput<Type>**

Generic Description: Type represents the underlying data that UIInput displays.

Class Description: UIInput is a base class for all boundary classes representing a server-client communication entity. It provides methods for gracefully deleting its instance, binding listeners, disabling, and re-enabling the UIInput.

Class Data Members: Type data, HTMLElement el, HTMLElement blocker, [HTMLElement, Function, string][] listeners

Class Methods: clickListen(HTMLElement, Function[], Boolean), disable(), enable(), destroy()

**CafeComment**

Class Description: CafeComment is responsible for rending a Comment communication entity.

Extends: UIInput<Comment>

Class Data Members: HTMLDivElement Username, HTMLDivElement Content, CafeCommentVoter funny, CafeCommentVoter factual, CafeCommentVoter agree, HTMLTextArea replyText, HTMLTextArea reportText, HTMLButtonElement submitReplyButton, HTMLButtonElement submitReportButton

Class Methods: replyClicked(), submitReplyClicked(), reportClicked(), submitReportClicked()

CafeCommentVote

Class Description: CafeCommentVote is responsible for rendering a CommentVoteDimension communication entity.

Extends: UIInput<CommentVoteDimension>

Class Data Members: number commentId, string voteType, HTMLDivElement voteLabel, HTMLButtonElement up, HTMLButtonElement down, HTMLDivElement total

Class Methods: upVoteClicked(), downVoteClicked()

**CafeMessageDisplay**

Class Description: CafeMessageDisplay is responsible for rendering a Message communication entity.

Extends: UIInput<Message>

Class Data Members: HTMLDivElement displayedText

Class Methods: updateMessage(Message), clearMessage()

**CafeUserDisplay**

Class Description: CafeUserDisplay displays UserProfile communication entity data with HTML Elements.

Extends: UIInput<UserProfile>

Class Data Members: HTMLDivElement Username, HTMLDivElement CreatedOn,HTMLDivElement DomainsModerating, HTMLDivElement IsAdmin, HTMLDivElement IsDomainModerator, HTMLDivElement IsGlobalModerator, HTMLDivElement ProfileBlurb

Class Methods: changeProfile(UserProfile), hide(), show()

**CafeFeedbackDisplay**

Class Description: CafeFeedbackDisplay displays a feedback record for an admin to view.

Extends: UIInput<FeedbackRecord>

Class Data Members: HTMLDivElement Content, HTMLDivElement Hide, HTMLDivElement SubmittedAt, HTMLDivElement FeedbackType, HTMLDivElement Username, HTMLButtonElement ToggleHidden

Class Methods: hideClicked()

**CafeOwnProfileDisplay**

Class Description: CafeOwnProfileDisplay is used to display a User's own profile to them.

Extends: UIInput<UserProfile>

Class Data Members: HTMLDivElement Username, HTMLDivElement CreatedOn,HTMLDivElement DomainsModerating, HTMLDivElement IsAdmin, HTMLDivElement IsDomainModerator, HTMLDivElement IsGlobalModerator, HTMLDivElement ProfileBlurb, HTMLButtonElement editProfileBlurbButton, HTMLTextAreaElement editProfileTextarea, HTMLButtonElement editProfileSubmitButton, HTMLButtonElement changePasswordButton

Class Methods: updateProfile(UserProfile), editProfileBlurbClicked(), editProfileSubmitClicked()

**CafeBanRecordDisplay**

Class Description: CafeBanRecordDisplay displays an individual ban or unban record.

Extends: UIInput<BanRecord>

Class Data Members: HTMLDivElement BannedFrom, HTMLDivElement BannedUsername, HTMLDivElement BannedByUsername, HTMLDivElement SetBannedTo, HTMLDivElement BannedAt, HTMLDivElement Reason

Class Methods: bannedUserClicked(MouseEvent) ,bannedByClicked(MouseEvent)

**CafeCommentReportDisplay**

Class Description: CafeCommentReportDisplay displays data for a single CommentReport.

Extends: UIInput<CommentReport>

Class Data Members: HTMLDivElement ActionTaken, CafeComment CommentData, HTMLDivElement ReasonReported, HTMLDivElement ReportingUsername, HTMLDivElement TimeReported, HTMLButtonElement moderateButton, HTMLInputElement setHiddenTo, HTMLInputElement setRemovedTo, HTMLTextAreaElement reason, HTMLButtonElement submitModerationButton

Class Methods: reportingUserClicked(MouseEvent) ,moderateButtonClicked(), submitModerationButtonClicked()

**CafeLogDisplay**

Class Description: CafeLogDisplay displays data from a single AdminAccessLog

Extends: UIInput<AdminAccessLog>

Class Data Members: HTMLDivElement Ip, HTMLDivElement Url, HTMLDivElement Username

Class Methods: usernameClicked(MouseEvent) ,

**CafeModRecordDisplay**

Class Description: CafeModRecordDisplay displays data from a ModerationRecord.

Extends: UIInput<ModerationRecord>

Class Data Members: CafeCommentReportDisplay AssociatedReport?, HTMLDivElement ModeratorUsername, HTMLDivElement Reason, HTMLDivElement SetHiddenTo, HTMLDivElement SetRemovedTo, HTMLDivElement TimeModerated

Class Methods: moderatorUsernameClicked(MouseEvent)

**CafeGlobalModDisplay**

Class Description: CafeGlobalModDisplay renders data from a GlobalModeratorRecord.

Extends: UIInput<GlobalModeratorRecord>

Class Data Members: HTMLDivElement GrantedAt, HTMLDivElement GrantedByUsername, HTMLDivElement GrantedToUsername

Class Methods: grantedByUsernameClicked(MouseEvent), grantedToUsernameClicked(MouseEvent)

**CafeDomainModDisplay**

Class Description: CafeDomainModDisplay displays data for a DomainModeratorRecord.

Extends: UIInput<DomainModeratorRecord>

Class Data Members: HTMLDivElement Domain, HTMLDivElement GrantedAt, HTMLDivElement GrantedByUsername, HTMLDivElement GrantedToUsername

Class Methods: grantedByUsernameClicked(MouseEvent), grantedToUsernameClicked(MouseEvent)

**CafeDomainReportDisplay**

Class Description: CafeDomainReportDisplay displays data from an AdminDomainReport object.

Extends: UIInput<AdminDomainReport>

Class Data Members: HTMLDivElement Domain, HTMLDivElement CommentCount, HTMLButtonElement xButton

Class Methods: xButtonClicked()

**CafeUsersReportDisplay**

Class Description: CafeUsersReportsDisplay displays data from an AdminUsersReport object.

Extends: UIInput<AdminUsersReport>

Class Data Members: HTMLDivElement LoggedInUserCount, HTMLDivElement NewestUsername, HTMLDivElement UserCount

Class Methods: newestUserClicked(MouseEvent), update(AdminUsersReport)

**CafeCommentSortDisplay**

Class Description: CafeCommentSortDisplay shows the users comment viewing settings and allows the user to change them.

Extends: UIInput<CafeSettings>

Class Data Members: HTMLSelectElement sortBy,HTMLInputElement viewHidden,HTMLInputElement sortAscending

Class Methods: settingChange(), updateFromSettings(CafeSettings)