Plan.ly

CS 673
Team 1 - UNO
Iteration 1

Team Overview

- Team **UNO**
- Working to build Plan.ly, an event planning tool

Name	Role
George Wright	Team Leader
Vibhu Bhatia	Backup Team Leader
Aysha Zenab Kenza	Security Leader
Karen Sommer	Design/Implementation Leader
Zach Schandorf-Lartey	Configuration Leader
Chris Kulig	Requirements Leader
Matt Dowding	Quality Assurance Leader

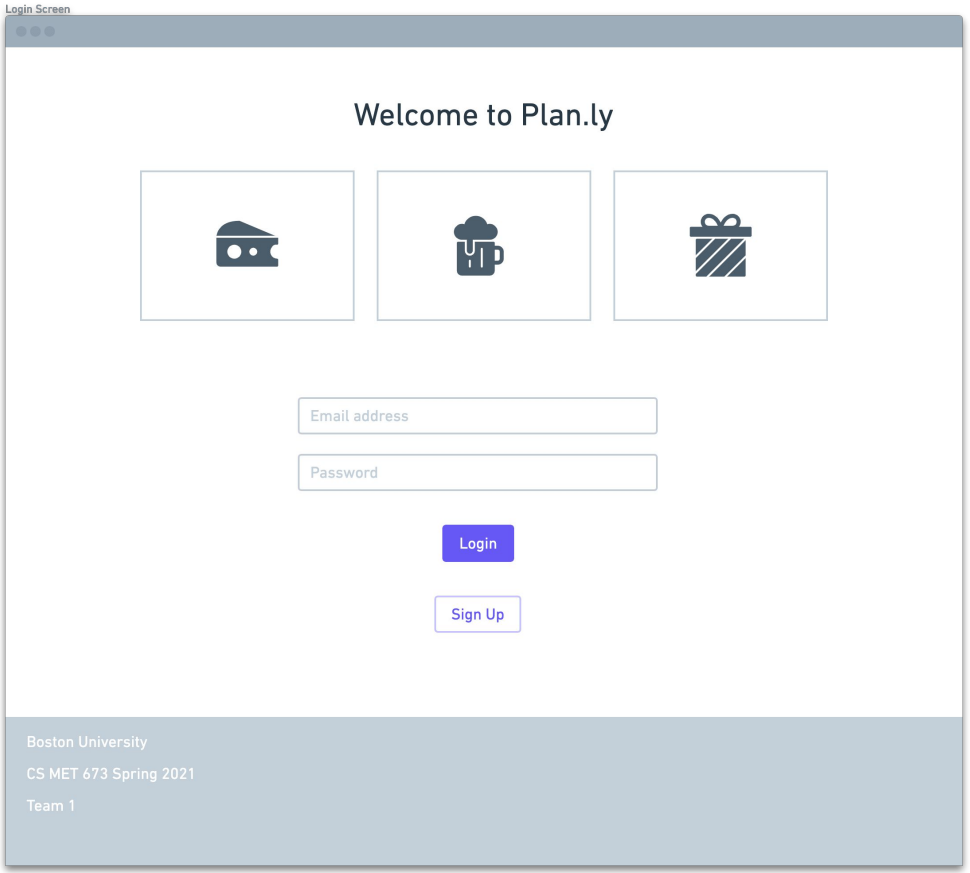
UNO

Introduction

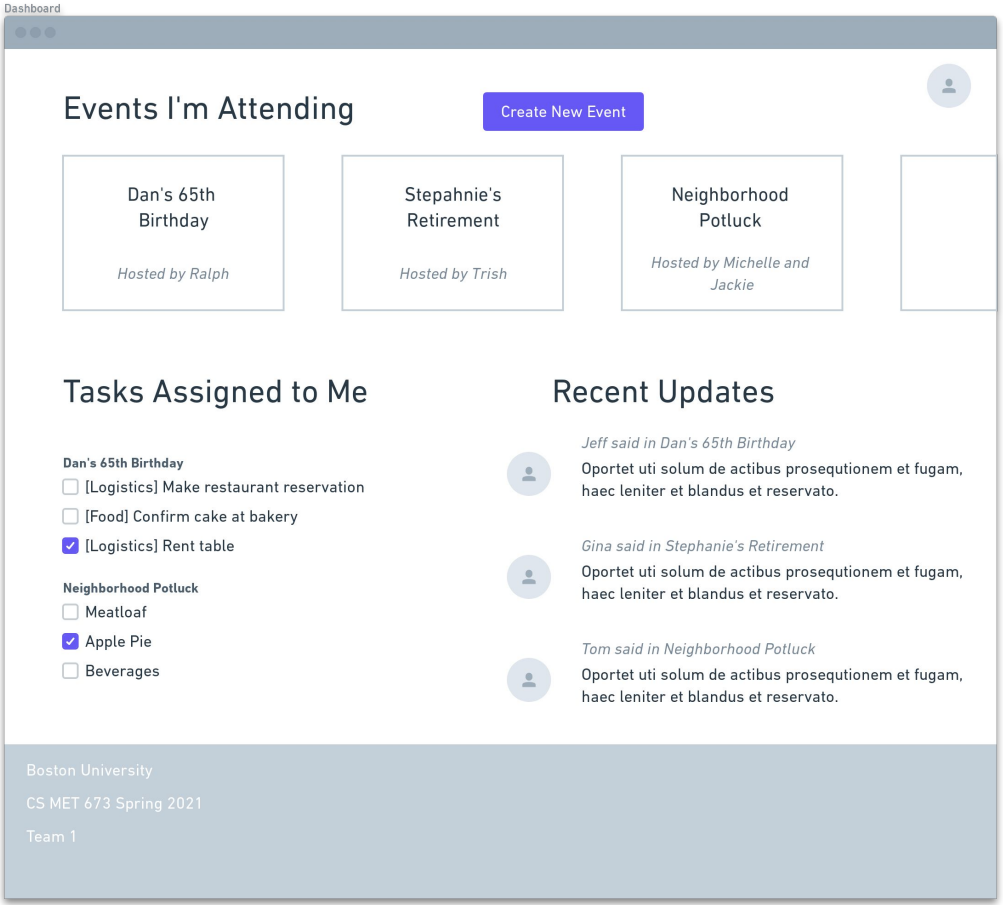
- Event planning application
- Essential:
  - Register, login
  - Create events, invite attendees, track to-do lists
  - Basic user dashboard/feed
- Desirable:
  - Comment feed within events
  - Enhanced user dashboard/feed
- Optional:
  - Photo upload
  - Email notification for task assignment

UNO

Homepage/Login Screen

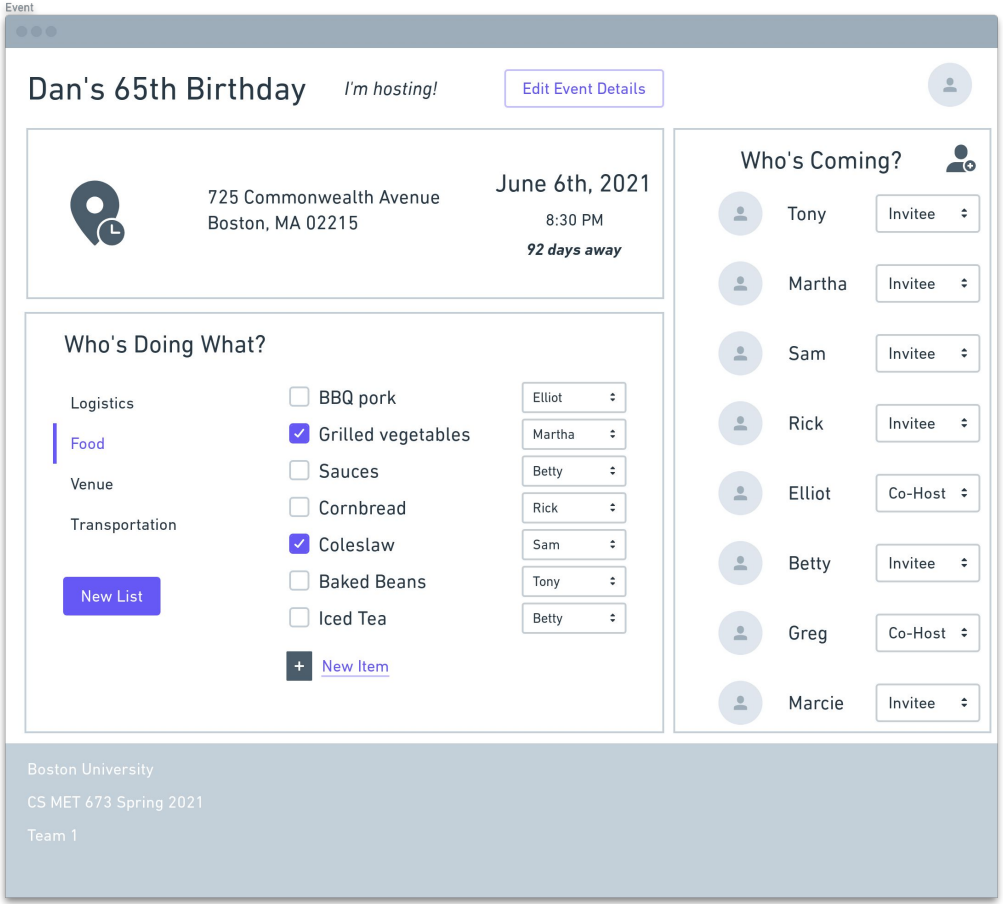


User Dashboard



UNO

Event Screen

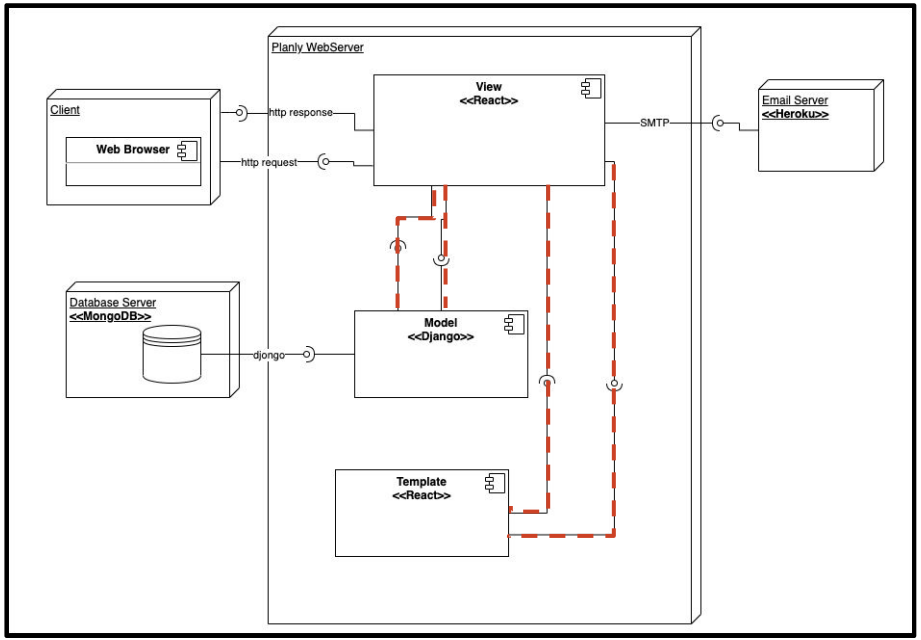


Software Architecture

Architectural Pattern

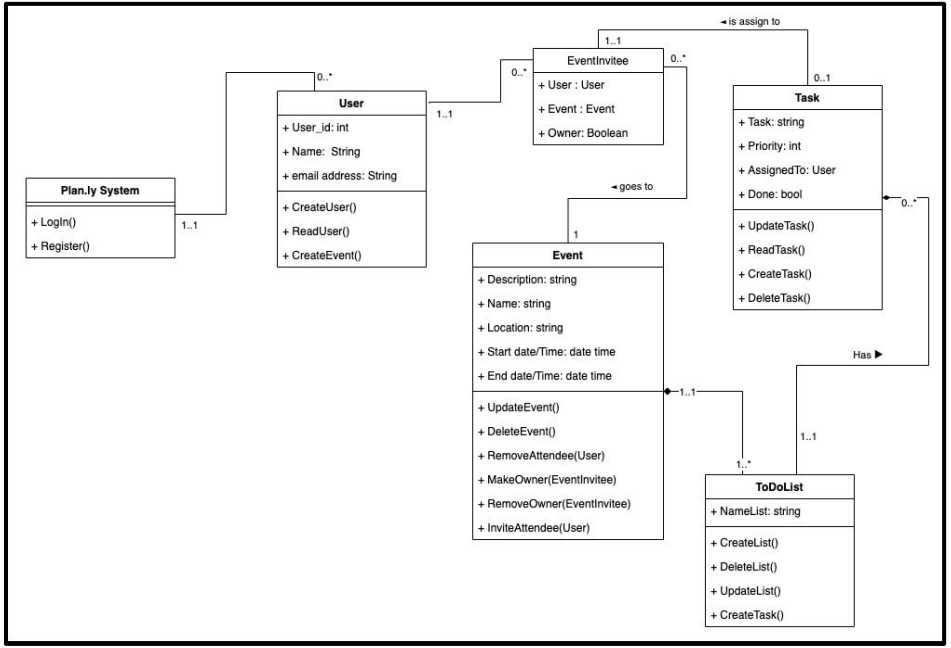
- Django architecture
- MTV pattern
  - Modified MVC.

-Component diagram



Software Architecture

-Class diagram



Classes

1. Plan.ly System
2. User
3. EventInvitee
4. Event
5. ToDoList
6. Task

Classes and Methods

UNO

Database

- Plan.ly uses NoSQL Database (mongoDB)
- Easy to set up, Simple to use, avoids expensive join operations.
- Allows Object Nesting for storing Data
- If one class contains instance of another, Only id is stored which gets populated with its class instance in runtime in near constant time.
- JSON data format universally accepted and parsed, allows for easy exchange between frontend and backend.
- Django Provides a mongoDB interface, Djongo which allows easy integration

Event Class

```
{
    "_id" : <DB generated id>,
    "Name" : "My Sample Event",
    "Description" : "This is the sample description for my awesome event. Please be there. I need help...",
    "Location" : "Boston, Massachussetts",
    "Start_date" : Jan 26 2021 10:20:00 AM,
    "End_date" : Jan 30 2021 10:20:00 AM,
    "Participants" : [Event_invitee1_id, Event_invitee2_id, Event_invitee3_id, Event_invitee4_id...],
    "Owner" : Event_invitee_id,
    "Lists" : [todoList_1_id, todoList_2_id, todoList_3_id, todoList_4_id..],
}
```

User Class

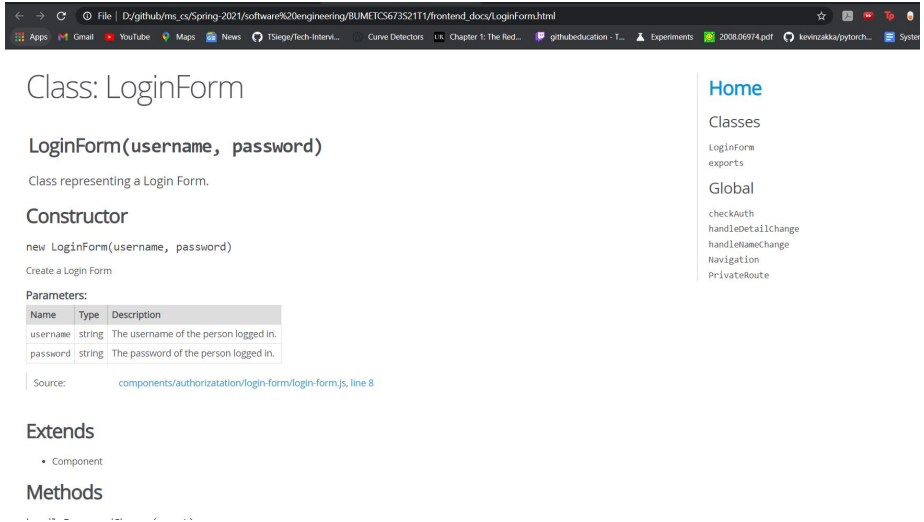
```
{
    "_id" : <DB generated id>,
    "user_name" : "Jon Doe",
    "password" : "$2y$12$6y874Qf4QaFvaGQMNa615ev",
    "email" : "Jon.doe@gmail.com",
    "first_name" : "Jon",
    "last_name" : "Doe",
}
```

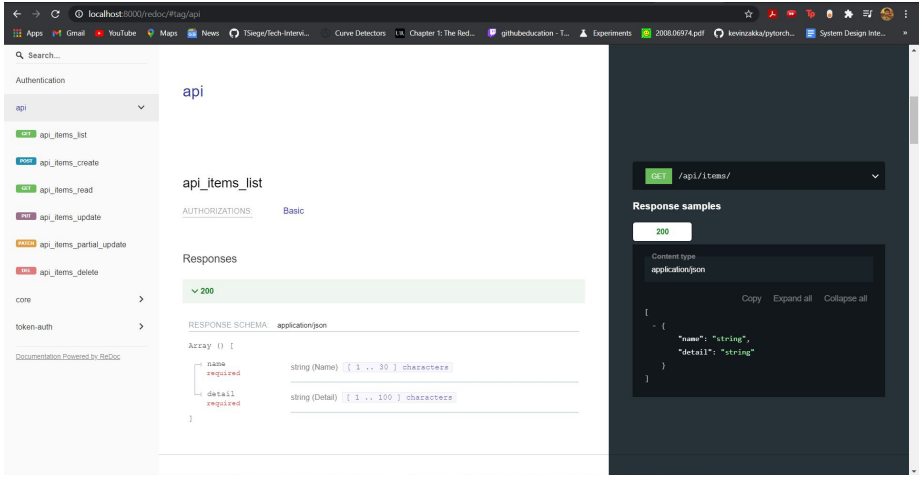
UNO

Classes and Methods

Backend : Use a framework called drf-yasg which generates real OpenAPI 2.0 specifications from Django rest API. [down-right]

Frontend : Use a framework called jsdoc which annotates the data in addition to generating automatic documentation for class methods. [down-left]





Security

- Cross Site Request Forgery (CSRF)
  - ➔ CSRF tokens are unique, secret, and randomly generated values that are generated by a server-side application and sent to the client.
  - ➔ When a request is made from the client, the server checks to see if the request includes the expected CSRF token, and if it does, validates it.
- Django
  - ➔ Django comes with a user authentication system. It handles user accounts, groups, permissions and cookie-based user sessions.
  - ➔ The auth system consists of:
    - ❑ Users,Permissions: Binary (yes/no) flags designating whether a user may perform a certain task,Groups: A generic way of applying labels and permissions to more than one user,A configurable password hashing system,Forms and view tools for logging in users, or restricting content,A pluggable backend system

- JSON Web Token
  - ➔ open standard that defines a securely transmitting information between parties as a JSON object, information can be verified using digital sign.
  - ➔ JSON Web Tokens consist of three parts separated by dots (.), which are:
    - ❑ Header,Payload,Signature

UNO

End-To-End Testing Cases

Test Case ID	Description	Type	Objective	Procedure	Expected Results
TC-01	Sign up with Plan.ly for the first time, using a valid email account	Positive Functionality	Ensure that Plan.ly allows new (unregistered) users to sign up using a valid email account	1. Open web browser and navigate to Plan.ly homepage 2. Click "signup" to initiate registration process 3. Choose the option to register with a Google account 4. Type in the credentials of a known email account to be used for testing 5. Click submit	1. Account is successfully created and user is redirected to the dashboard/feed page. 2. User receives an automated email (at the email address associated with the email account) confirming that the registration was succesful.
TC-02	Log in to Plan.ly with a valid email account	Positive Functionality	Ensure that Plan.ly allows a previously registered user to sign in with their associated email account	1. Open web browser and navigate to Plan.ly homepage 2. Click "Sign In" button to initiate logging in 3. Choose same email account the user has previously signed up with 4. Click "submit" button	1. Successful log in with correct Google account 2. Redirect to user dashboard
TC-03	Logout from Plan.ly	Positive Functionality	Ensure that Plan.ly allows users to logout after they successfully logged in	1. Open web browser and navigate to Plan.ly homepage. 2. Login with known email account credentials that were previously used to register on Plan.ly 3. Once logged in, click logout	1. Successfully logged out of users account 2. Return to Plan.ly homepage
TC-04	Logout from Plan.ly and attempt to click back button	Negative Functionality	Ensure that on successful log out a user cannot hit the back button to see restricted content	1. Open web browser and navigate to Plan.ly homepage. 2. Login with known email account credentials that were previously used to register on Plan.ly 3. Once logged in, click logout 4. Click the "back" browser button	1. The user is not able to return to previous restricted content 2. User redirected to Plan.ly homepage

UNO

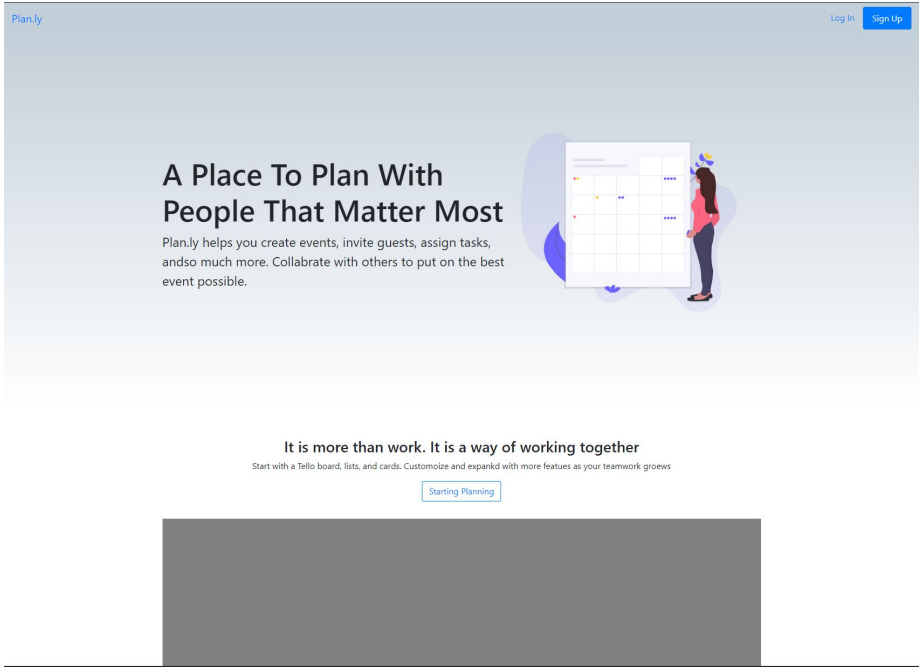
Requirement Testing Coverage

Id	Title	Description	Labels	TC-01	TC-02	TC-03	TC-04
176879523	Log in	As a web site/user I want to be able to log into my account to securely gain access to my resources. -2H	essential, user authentication/registration/profile				
176949922	Sign up	As a website user I want to be able to create an account to be able to create, view, and manage my events 8.5H	essential, user authentication/registration/profile				
176936657	Receive Account Creation Email Notification	As a web site user/member, when I register for a Plan.ly account using my Google credentials, I want to receive an email notification so that I can know my registration was succesful. - 6H	essential, user authentication/registration/profile				
176953068	Go To Event From Feed	As a logged in user on the home page when there are events available in the feed I want to be able to click on an event within the feed and go to the details page to get information about it not available through the feed.	desirable, user dashboard/feed				
176948560	Event owner delete subtask	As an event owner, I want to be able to delete any subtask on any task on any task list so that I can correct the list of subtasks for a particular task. - 1.5 h	optional features, optional-subtasks				

UNO

Demonstration

[Project demo]



Thank you

Questions?