Powell Puff 38

Divyam, John, Tung, Jason, Hugo,
Rachel, Thu, Nina, Vedant, Yousuf

TA: 💖 Allison 💖

App Description

Schedule Planner

- stores class times & details
  - finals schedule
  - possible extension for pulling from webreg
- tracks assignments
  - due dates, color coordinated (by class)
  - tools for scheduling out work-time for assignments
- tracks quiz & exam times
  - reminders to study for exams
  - don’t be surprised week 11 what final you have first!

Implementation Details

Product:

- Hosted site on GitHub Pages
- HTML/CSS user interface
  - Single page with tabs.
  - Individual *views* manipulated via javascript for live updating of elements and data access.
- Database: .json storage implementation in browser storage
  - Exportable for archiving.
  - Upgraded to SQL styles database if time allows.

Minimum Viable Product

- no importing schedule from WebReg
- one working view (Today View)
  - no live updating (refresh)
- classes & tasks in appropriate locations
  - no detail views
- fixed time window (6A-12A)
- tasks show up
- manually add information in settings window.
  - With automatic repeating for each week for classes.
  - Tasks assigned to classes

Nice to have features (if time permits)

- 4 different, fully functional views
- Live-updating UI
  - “now” line
- WebReg Parsing
- Time Blocking assisting
  - for exam/finals studying
  - for completing assignments
- More database integrations
- Desktop notifications
- Export/Import
  - Class schedule archiving
  - .ics calendar files
- Support for mobile browsers
- Time zones

User Experience

Setup:

- Import schedule from webreg HTML
- Manually add labs, internships, etc.

Usage:

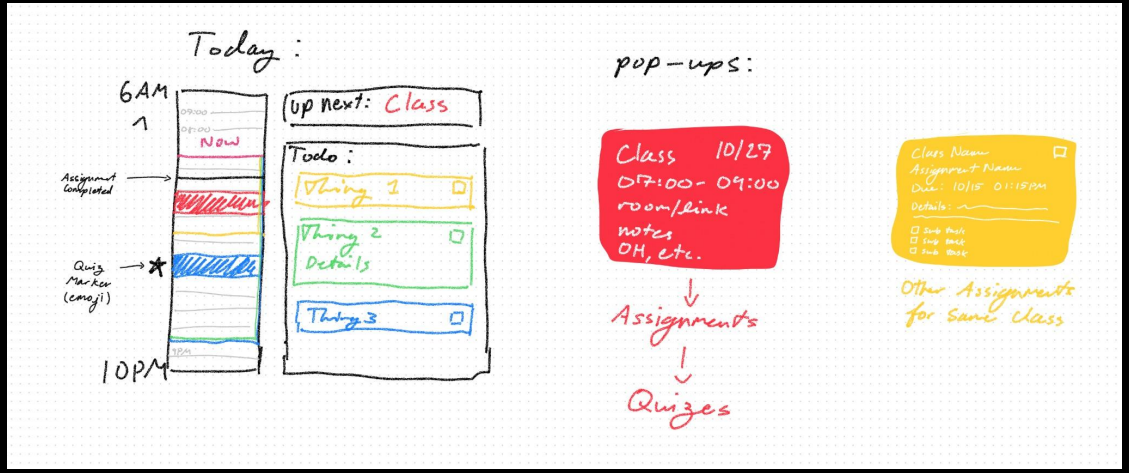
- Add assignments for each class as they become available.
- Today view shows what to work on next.
- Close all your colors (complete assignments) to track your work for the day.

Visual Design

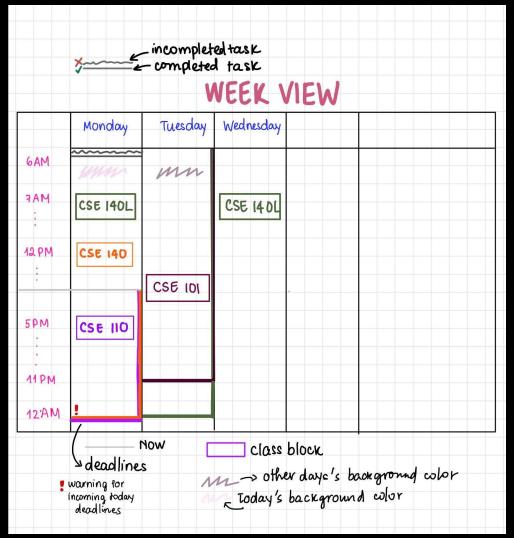
Use color to track class times & assignments.

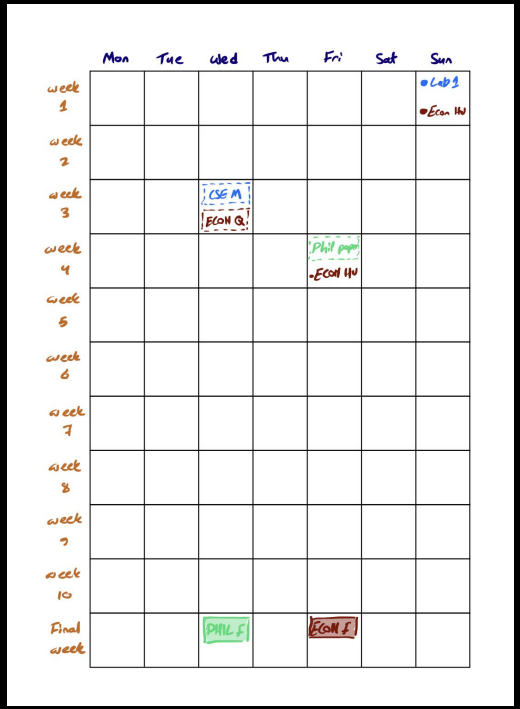
Different Views for organizing information:

- Today view
- Week view
- Quarter view
- Finals view
- Tasks



Week View, Quarter View, Finals View





Final View

	Sat	Mon	Tue	Wed	Thu	Fri	Sun
6am							
7am							
8am		CSE 101 Final exam					
.							
.							
12am	CSE 100 Final exam						
1pm							
2pm					CSE 110 Final project		
.							
.							
.							
10pm							

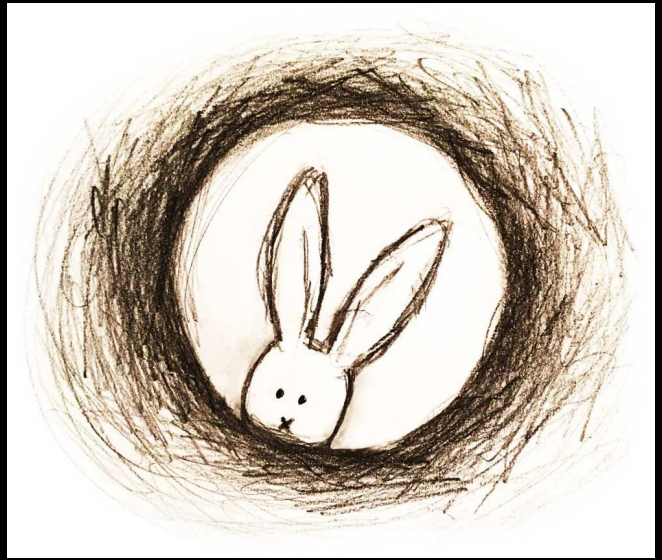
Risks & Rabbit Holes

Project:

- Short, 4 Week Timeline.
- High learning curve for web development (little prior experience)
- Large initial feature base (will likely need to cut things)

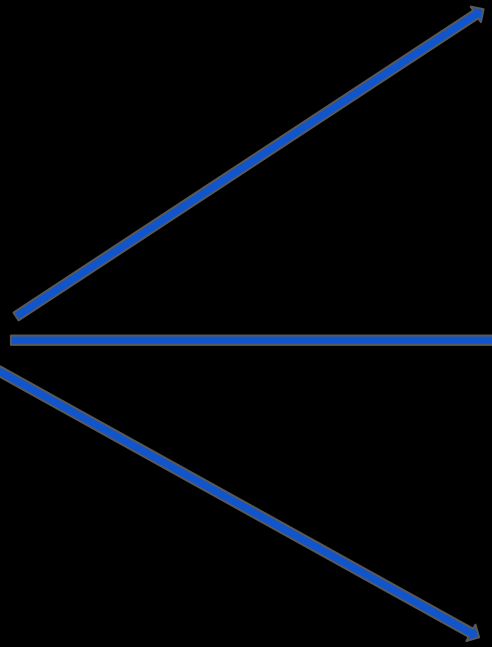
Technical:

- Complex & Live UI may be difficult to implement
- Parsing webreg information may be difficult.
- Complex information relationships to manage in database.

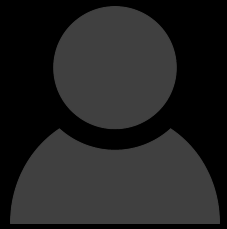


Personas

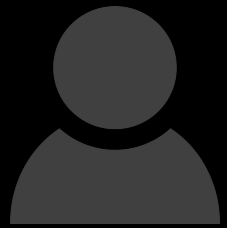
USERS
TYPES



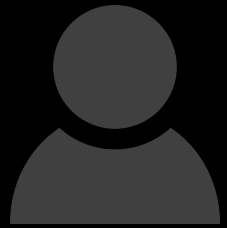
Student



Professor



TA/IA



User Persona #1 (Student/TA)

Name: Charlotte Johnson

Age: 21

Occupation: Undergraduate student and TA

Motivations: Achievement

Growth

Goals: Motivated

Loves being organized

Wants to get done with deadlines as early as possible

Does not like forgetting about work due

Wants to pre-plan her days for peace of mind, and stress and time management

Frustrations: Forgetful

Does not check phone too often if there are no notifications

User Persona #2 (Professor)

Name: Thomas Powell

Age: 54

Occupation: Professor

Motivations: Achievement

Goals: Manage timelines and meetings with a lot of students,

TAs and other faculty

Loves being organized

Wants to have a good sense of the deadlines assigned

Wants to ensure smooth running of a course schedule

Wants to pre-plan his days for peace of mind, and stress and time management

Frustrations: Does not check phone too often if there are no notifications

Has too many people’s schedules to keep in mind apart from him own

Database Design

Information we need to store

Classes: (defines basic information for recurring events)

- association (class-code, “personal”, etc.) “”
- final Date & Time
- class times array [time]
- lab time
- discussion time

Events: (stores basic calendar information for specific occurrences of events)

- start & end time
- weekday 0-6
- event name “”
- association “”
- exam dates [“”]

Tasks: (stores due-date information for todo-type things)

- task type (todo, assignment, subtask) “”
- due date
- association “”
- description “”
- link “”
- task name “”
- subtasks [“”]

Database Design

Schema

- Users (UUID VARCHAR(50) PRIMARY KEY, Password_Hashed VARCHAR(50), USER_CREATION_DATE Datetime)

- USER_INFORMATION (UUID VARCHAR(50) Foreign Key, EVENT_CREATION_DATE Datetime, EVENT_TYPE VARCHAR(50), EVENT_DATE Datetime, EXPIRE_DATE Datetime, EXPIRED Bool)

EVENT_TYPE :

- CLASS
- EXAM_MIDTERM
- EXAM_FINAL
- DI
- LAB
- REMINDER

Database Design

USERS	
UUID	VARCHAR(50)
Password	VARCHAR(50)
User Creation Date	DATETIME

USER_INFORMATION	
UUID	VARCHAR(50)
Date_Added	DATETIME
Event_Type	VARCHAR(50)
Event_Date	DATETIME
Expire_Date	DATETIME
Expired	BOOL

Statement of Purpose

Stress free, time efficient and organized

Save time and mitigate stress for students:

- Reduce friction for building schedules
- Tracking deadlines and upcoming exams to assist in blocking work-time.
- Get a clear picture of a day before starting.

Timeline & Roadmap

1. **Week 5:** Formation of groups and role assignments.
2. **Week 6:** Planning + Early development
   - Deciding on proper tools for development and debugging.
   - Deciding on the documentation style.
   - Build basic app structure.
   - Start working in groups.
3. **Week 7:** Pages setup (if not partially functional)
   - data entry and saving working (.json)
   - demo of today-view working.
4. **Week 8:** Finish Core Features
   - Proper operation of classes & tasks
   - Have minimum viable product in place
5. **Week 9:** Border features (outside of MVP)
   - Implementing extended features if possible.
   - Testing and debugging UI and MVP.
6. **Week 10:** Finalize documentation.
   - Throw a party if we get an A on the project.