Procrastify - Group 6

Alexander Ho, Megan Ingram, Michael Kasman, Samah Khan, Alan Liao, Damian Ozuna, Sarah Tempelmeyer

Objective

- Procrastify, a mobile app that organizes and displays homework, quizzes, tests, etc. in an agenda format with to-do list and note taking capabilities
- Procrastify implements the concept of productive procrastination to aid students in completing tasks, A digital agenda suits our needs of being informed on what we should be working on.

Cost Estimation

- Estimated cost of hardware products:
  - Work Machine Budget: $800/per machine
  - Work Machine Total: $800 x 6 person team = **$4,800**

  - Amazon Web Services Server Cost
  - Number Users = 100,000 users
  - Average Number of User Activities per day = 5 activities
  - Total Activities/Log Entries per day = 100,000 x 5 = 500,000
  - Approximate Log Entry Size = 500 bytes
  - 500 bytes x 500,000 activities = 250 Million bytes of data = 2.5 GB of log data per day(scaled by a factor of 10 for worst case)
  - Log Data per month: 75 GB
  - Amazon S3 Data Storage Cost = $0.23/per GB
  - Total Cloud Storage Cost = $0.23 x 75 GB = **$17.25 per month**

Cost Estimation

- Estimated cost of software products:
  - Auth0 login authentication - For 10,000 active users: $228 per month
  - Jira Software Standard Package - $70 per month
  - Amazon Simple Notification Service - $0.50 per million notifications

- Estimated cost of personnel:
  - 6 Person Development Team
  - 3 Software Engineers/Backend Developers
  - 2 Front-End Developers
  - 1 Database Administrator
  - 4 Month Contract
  - Salary = $50/hr = $2,000/week = $8,000/month
  - Total Personnel Cost = $8,000 x 4 months = **$32,000**

Project Timeline

Start Date: Jan 4, 2021

Task	Duration (weeks)	Deadline	Endpoint
Design Interface, Software, & Database	3	Jan 22, 2021	Processes, tools, classes, UI/UX plan, database security plan, produce design reports
Create Prototype	2	Feb 5, 2021	Prototype
Develop System Modules	6	Mar 19, 2021	Interface, software, and database fully developed
Testing and Deployment	5	Apr 23, 2021	Testing and hosted application

E = FP/productivity
= 187.2 FP / 30 FP per person-weeks
= 6.24 weeks or 44 working days.
Approximately 44 days will be allocated for each component of the project:
1) design/prototyping
2) development
3) testing/deployment.

Start Date- Jan 4, 2021

Task	Duration (weeks)	Deadline	Endpoint
Design Interface	2	Jan 15, 2021	Application design and UI/UX
Design Software	2	Jan 15, 2021	Processes, classes, and tools
Design Database	2	Jan 15, 2021	Processes, tools, information security
Create Design Specifications	1/2	Jan 20, 2021	Report on specifications
Complete Design	3	Jan 22, 2021	Design
Complete Prototype	2	Feb 5, 2021	Prototype
Develop System Modules	6	Mar 19, 2021	Interface, software, and database developed
Create Development Specifications	1/2	Mar 19, 2021	Report on specifications
Complete Development	6	Mar 19, 2021	Developed application
Perform System Testing	2	Apr 2, 2021	Testing successes and failures
Correct System Errors	2	Apr 16, 2021	Tested application
Deploy	1	Apr 23, 2021	Public application on server

End Date: Apr 23, 2021

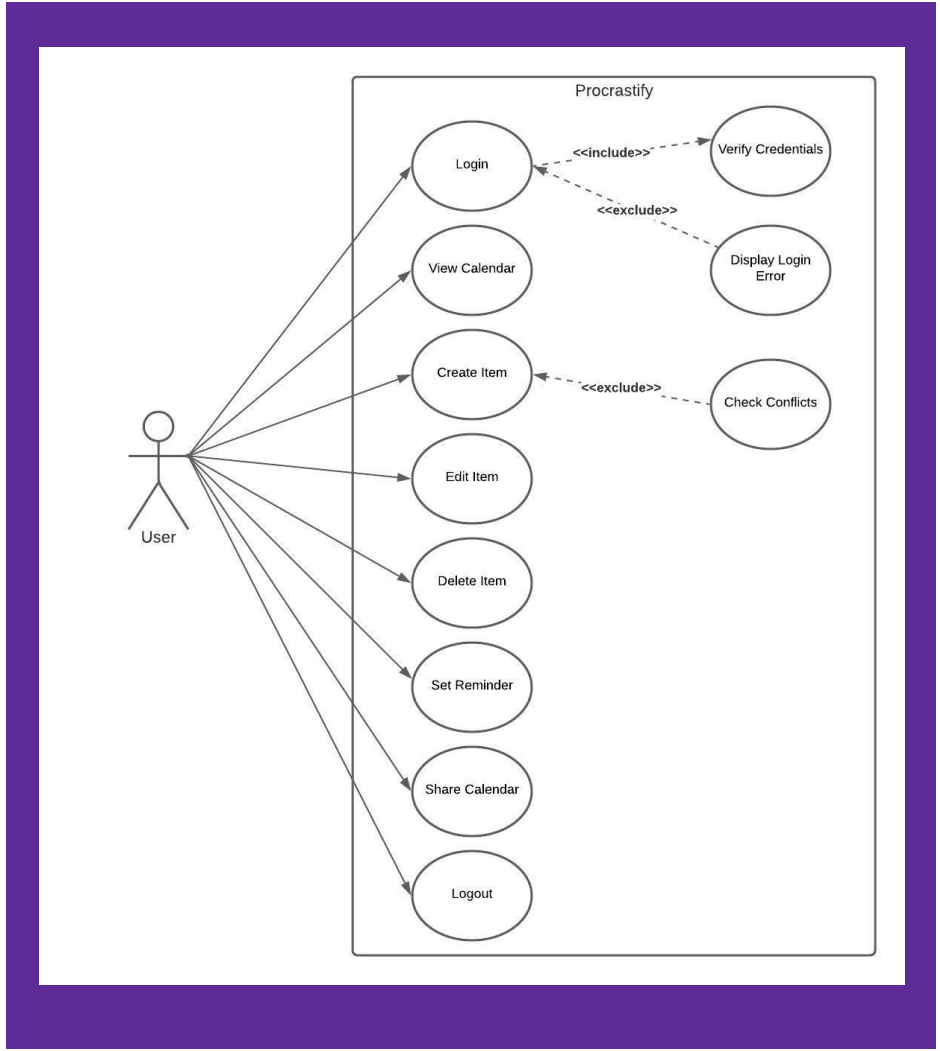
Functional Requirements

- A user should have a unique email and password to both identify themselves and login with
- A user should have a weekly/monthly/yearly view, agenda view, to-do list view, and a notes view in a calendar display with all items entered by said user
- A user should be able to enter an item with a title, content, category, and date details and have it stored on their calendar
- A user should be able to search their calendar for all items entered by date, title, category, and course(if provided by user)
- A user should be able to edit or delete any chosen item from their calendar
- The user should receive notifications from the system for upcoming items at intervals selected by the user
- A user should be able to share their calendar through the email provided during login

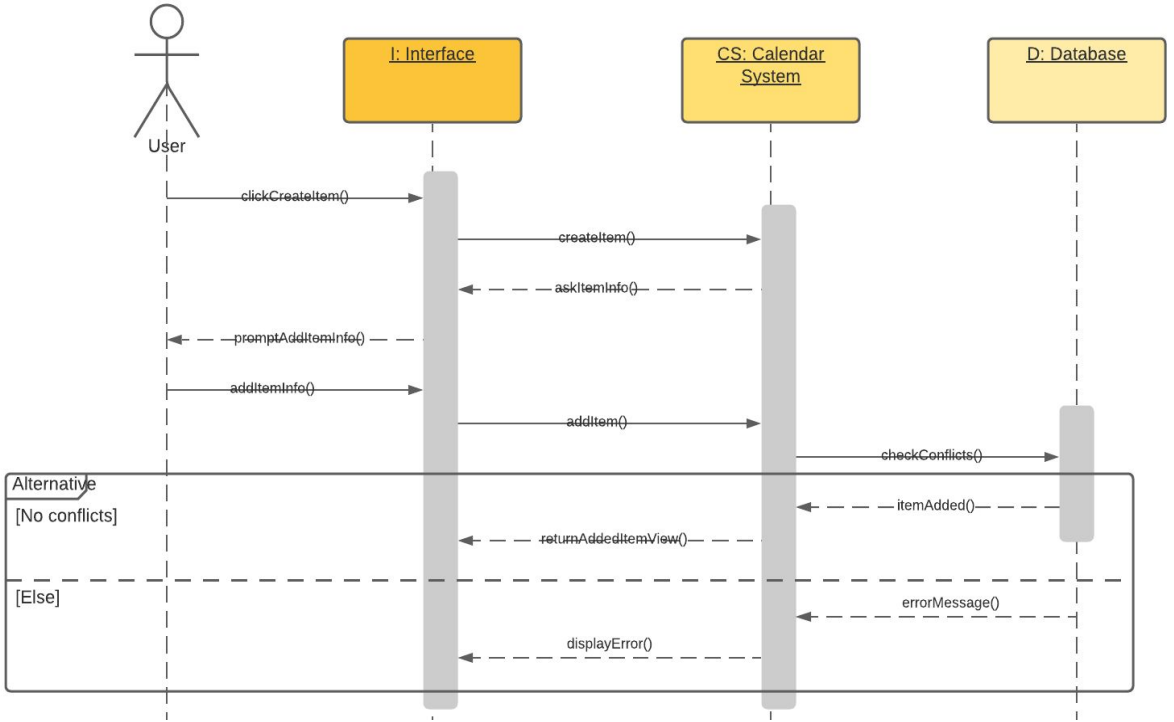
Non-functional Requirements

- Usability Requirements: The application must be intuitive enough to navigate through the calendar as well as its functions with ease. The user should be able to search, create, delete, or edit objects from the calendar display.
- Performance Requirements: Application must be efficient enough to operate smoothly on mobile devices by maintaining a relatively small footprint in terms of CPU and memory usage.
- Space Requirements: Application must be compact enough to be stored on limited space of new and older mobile devices. Ideally the application will be smaller than 500 MB.
- Dependability Requirements: The system should reliably provide its core functionality of creating objects and organizing the data within them even when offline.
- Security Requirements: The application must authenticate the user's account before granting access to data stored on the app. This can be implemented in different ways, such as touch or face ID as well as through a password.
- Operational Requirements: The system will be used to automatically organize objects by date and type (Homework, quizzes, notes). These objects are initialized with data given by the user and are stored by the application on the user's device.
- Development Requirements: This application will be written using Java as an Android app.

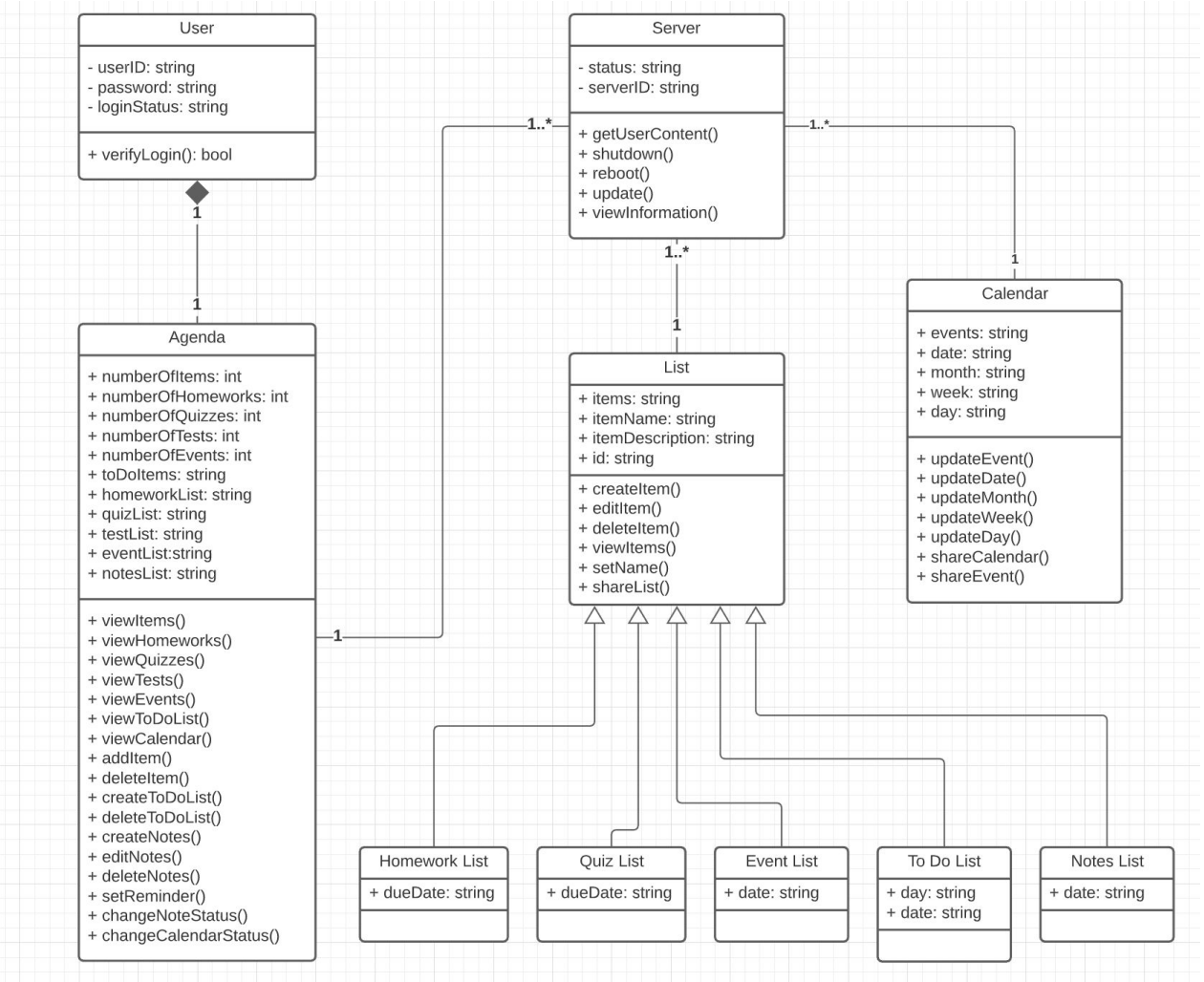
Use Case Diagram



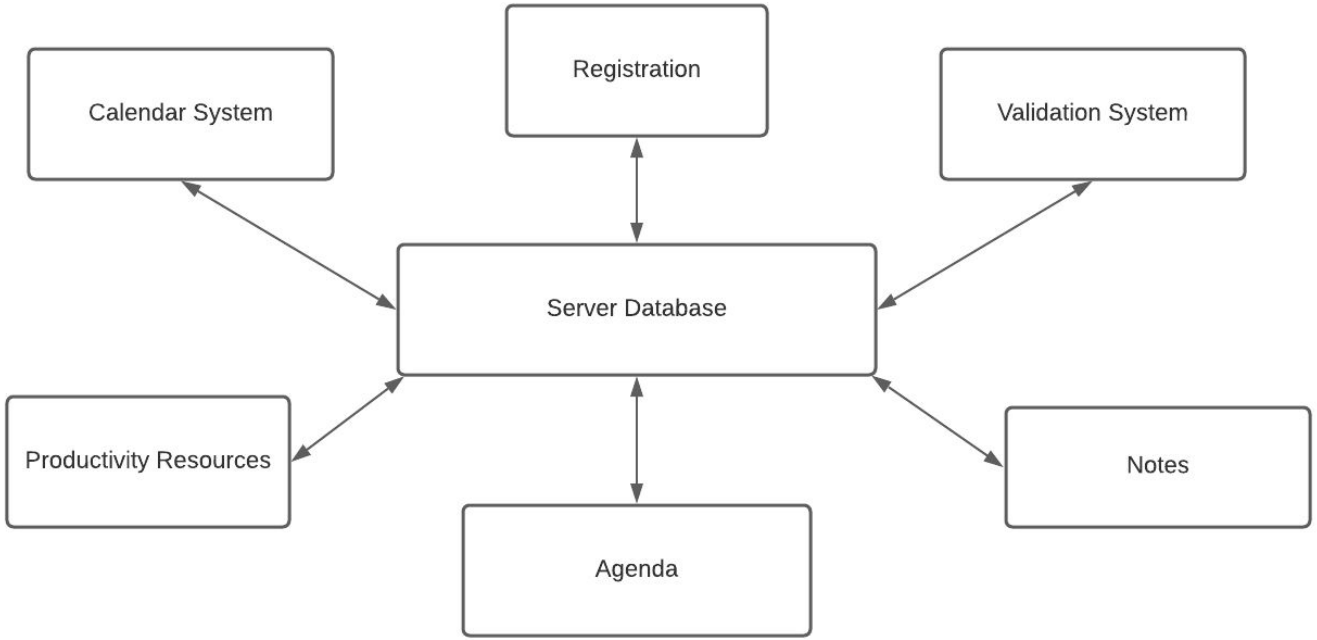
Sequence Diagram



Class Diagram



Repository Architecture Pattern



User Interface Demo

Let’s take a look at the interface of Procrastify.

Procrastify

Welcome Back!

Login

Email

Password

Sign In

Don't have an account? Sign Up

Procrastify

Stay on top of your work and yourself with our customizable features and helpful resources when you sign up today!

Have an account already? Log in!

Sign Up

FIRST NAME

LAST NAME

EMAIL

PASSWORD

CONFIRM PASSWORD

Login Register

Logout

Procrastify

Calendar | Notes | Profile

Your Categories

- Exam
- Quiz
- Add

Change View

7

Friday

Event

To-do List

Note

Reminder

2020

Jan Feb Mar Apr May Jun July Aug Sep Oct Nov Dec

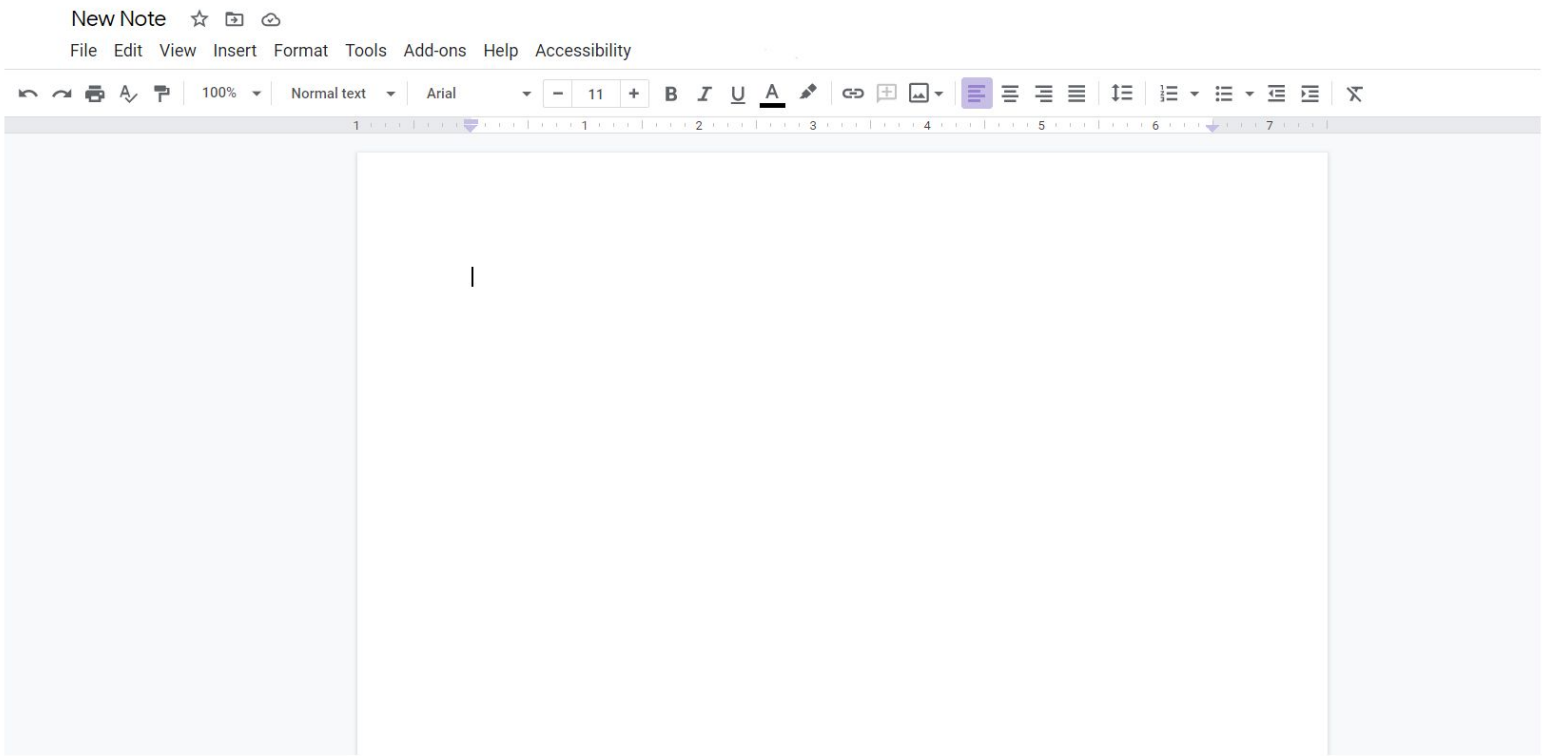
SUN	MON	TUE	WED	THURS	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Logout

Procrastify

Calendar Notes Profile

CS 3354
- Project
  - New Note



Logout

Procrastify

Calendar Notes Profile

Profile Settings

Name

User Example

Email

user@gmail.com

Password

••••••

Update

Competitor Comparison

Procrastify Extra Features:

- Agenda View
- Create Color categories
- To-do lists
- Notes

Procrastify Extra Features:

- Share Calendar
- Scrolling Agenda
- Notes

Procrastify Extra Features:

- Yearly and Agenda View
- Create color categories
- To-do lists
- Notes
- Reminders

Conclusion

- Main changes stem from how our product differed from already existing products
  - Added more flexibility around creation, organization, and displaying events to further set our product apart
  - Added the ability to organize by event type (homeworks, quizzes, exams, etc)
  - Added note-taking and courses as an identifier for sorting and planning
  - Added the ability to share schedules
- While small concepts were added, there was very little deviation from the app's original core idea
  - Each change enhanced the original idea and worked with the original capabilities to provide a better product

Q&A/Contact Information

- Alexander Ho - Alexander.Ho@utdallas.edu
- Megan Ingram - Megan.Ingram@utdallas.edu
- Michael Kasman - Michael.Kasman@utdallas.edu
- Samah Khan - ssk170006@utdallas.edu
- Alan Liao - Alan.Liao@utdallas.edu
- Damian Ozuna - Damian.Ozuna@utdallas.edu
- Sarah Tempelmeyer - Sarah.Tempelmeyer@utdallas.edu