CheckMate

By: Sanju Kanumuri, Hafsa Khan, Manas Keithreddy, Yoon Lee, & Nandini Yella

Original Problem Statement

- Centralized communication and task management specific to software professionals.

- Lack of unified platform integrating emails, chats, and other tools for streamlined workflows.

- Real-time collaboration with clear task organization and deadline synchronization.

- Enhanced team efficiency and reduced cognitive load through a single, user-friendly system.

Explanation and Rationale

- **Centralized Platform:** CheckMate merges communication and task management tools into one system.

- **Organized Tasks:** Facilitates task creation, categorization, and assignment for streamlined workflows.

- **Team Collaboration:** Enhances communication with features like task commenting and real-time chat.

- **Information Centralization:** Allows due dates and file attachments for easy access to task details.

- **Version Control Compliance:** Tailored for software teams, ensuring consistent project tracking.

Use Case Diagrams:

Three sample cases

Use Case: Creating A Task

1 Preconditions

User must be logged into their account

2 Main Flow

User will input a task and provide a description[S1].

3 Subflows

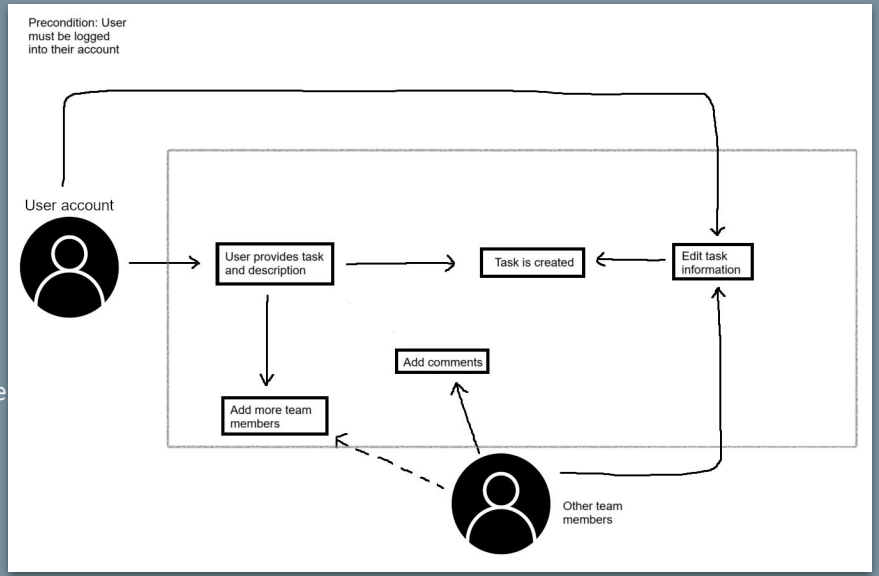
[S1] Users can add any additional team members to tasks.

[S2] Other users can see the created task and comme on it.

[S3] All users can discuss tasks and make adjustments.

4 Alternative Flows

[E1] No changes to tasks are needed.



Use Case: Integrate Code into Task

1 Preconditions

User must be logged into their account

2 Main Flow

User will select a task and click on “Integrate Code”[S1]. System will prompt for repository details[S2]. User provides details and confirms[S3]. System links the codebase to the task[S4].

3 Subflows

[S1] Users can add any additional team members to tasks.

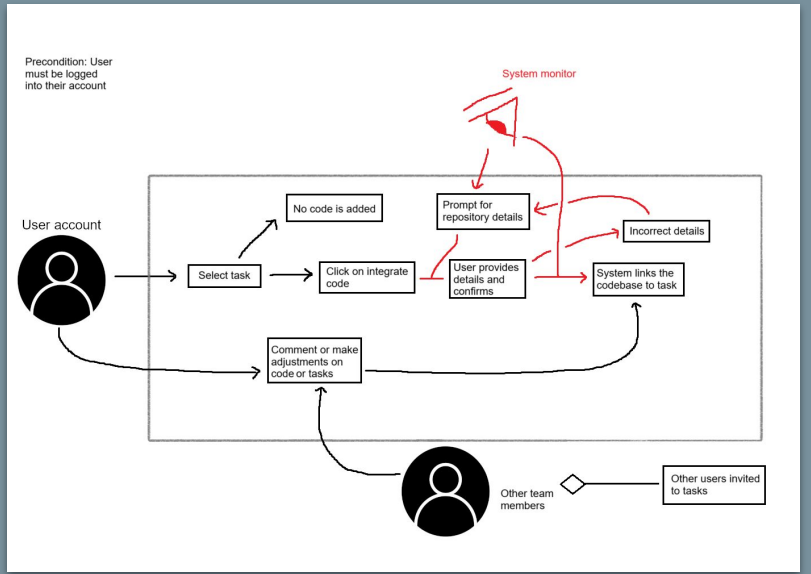
[S2] Other users can see the code and comment on it.

[S3] All users can discuss tasks and make adjustments.

4 Alternative Flows

[E1] No code is added to the task.

[E2] Right repository details were not entered so system reprompts.



Use Case: Engage in Real-Time Chat

1 Preconditions

User must be logged into their account.

User navigates to a specific task.

2 Main Flow

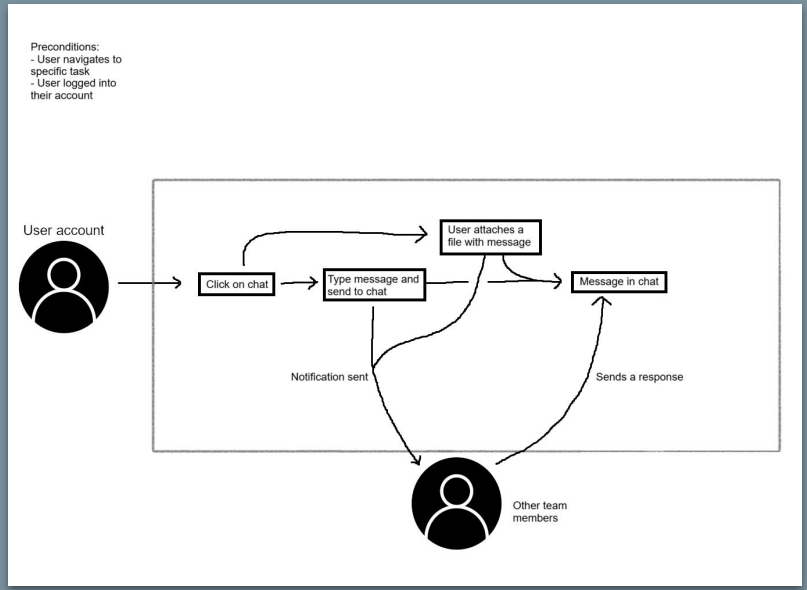
User will click on “Chat” associated with task[S1]. User types messages and sends[S2].

3 Subflows

[S1] Team members receive notification and can respond.

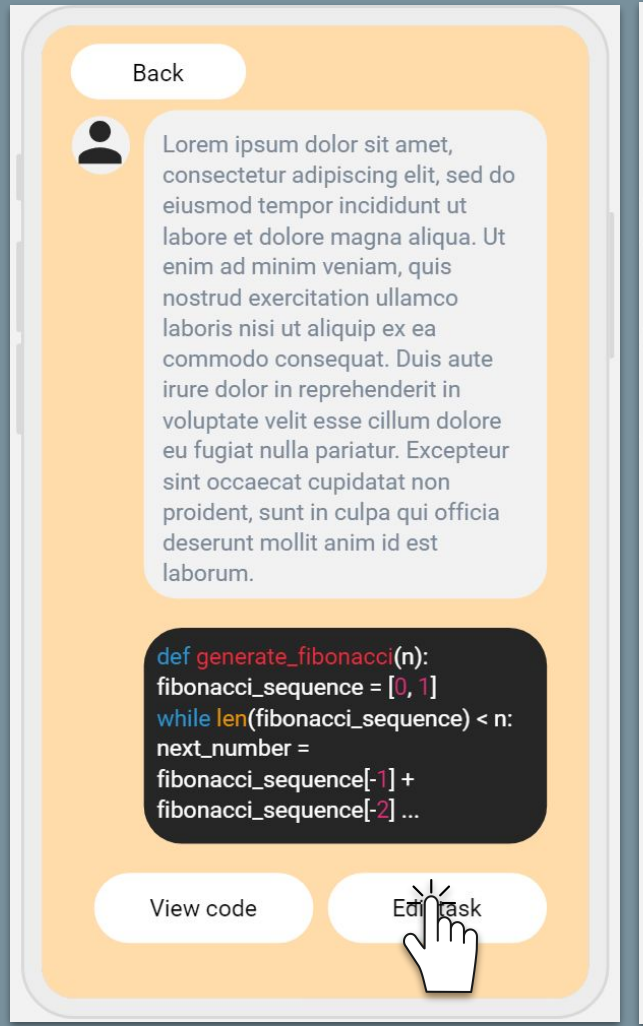
4 Alternative Flows

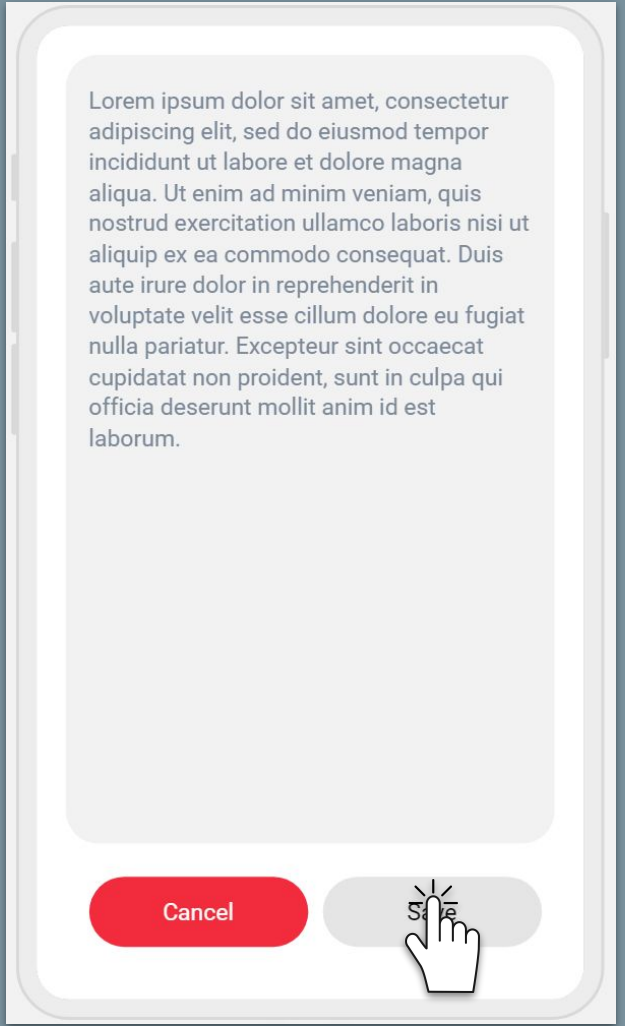
[E1] User can attach files in chat along with text.



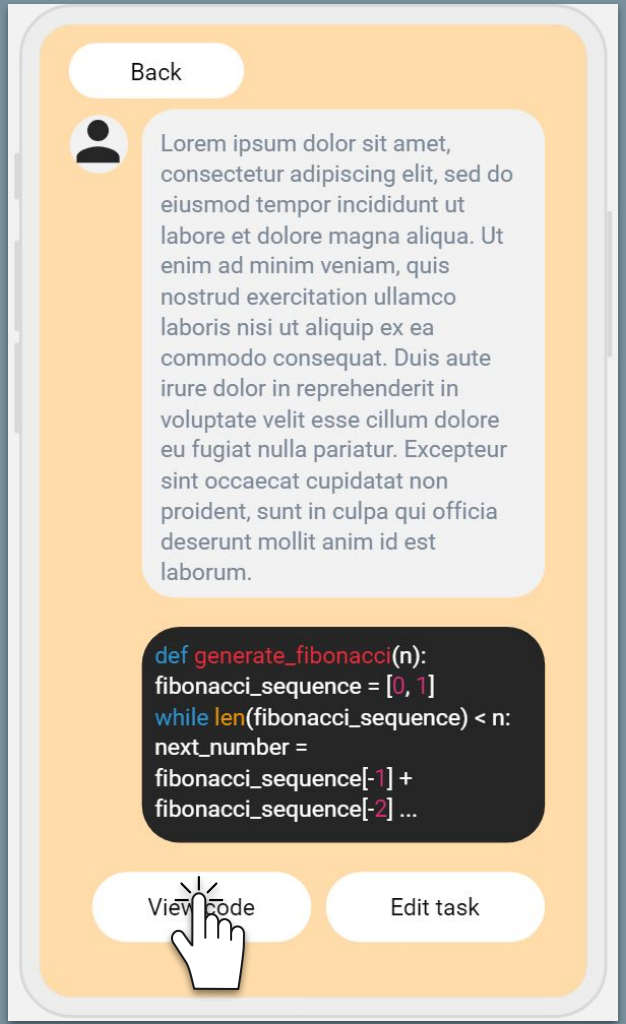
Visual Representation

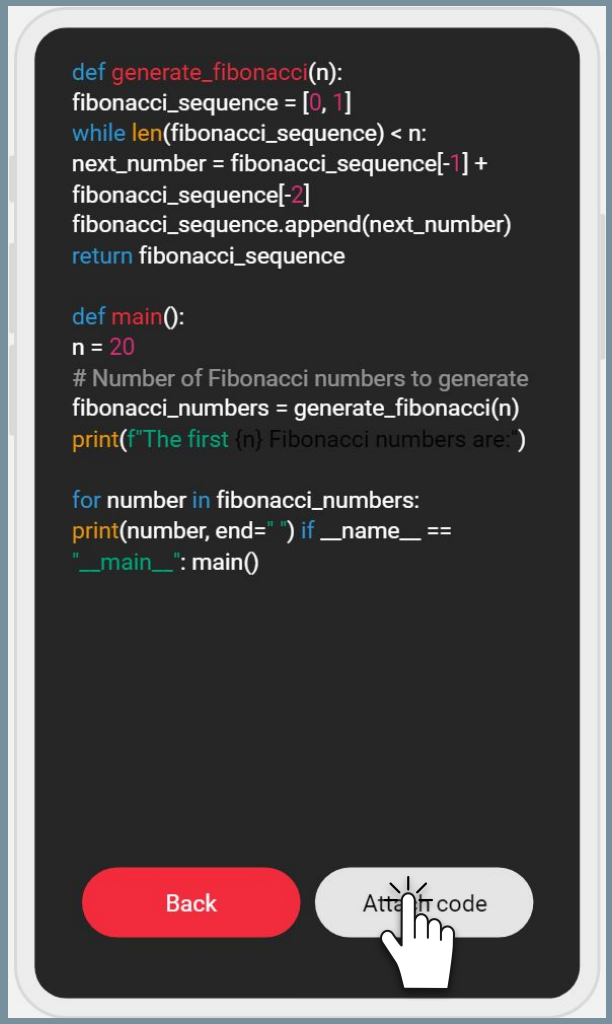

Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
...

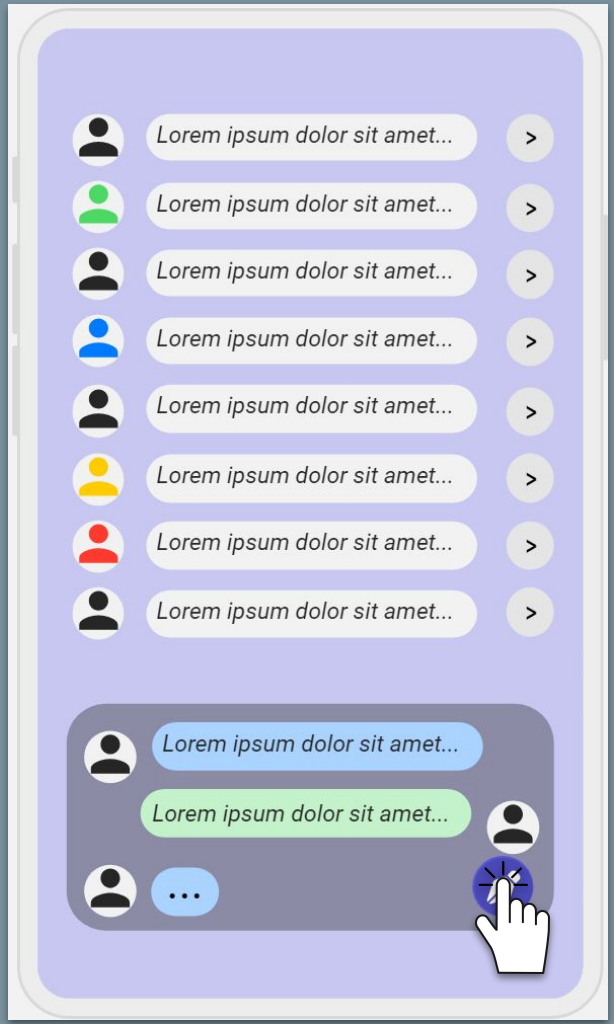

Back
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
def generate_fibonacci(n):
fibonacci_sequence = [0, 1]
while len(fibonacci_sequence) < n:
next_number =
fibonacci_sequence[-1] +
fibonacci_sequence[-2] ...
View code
Edit task

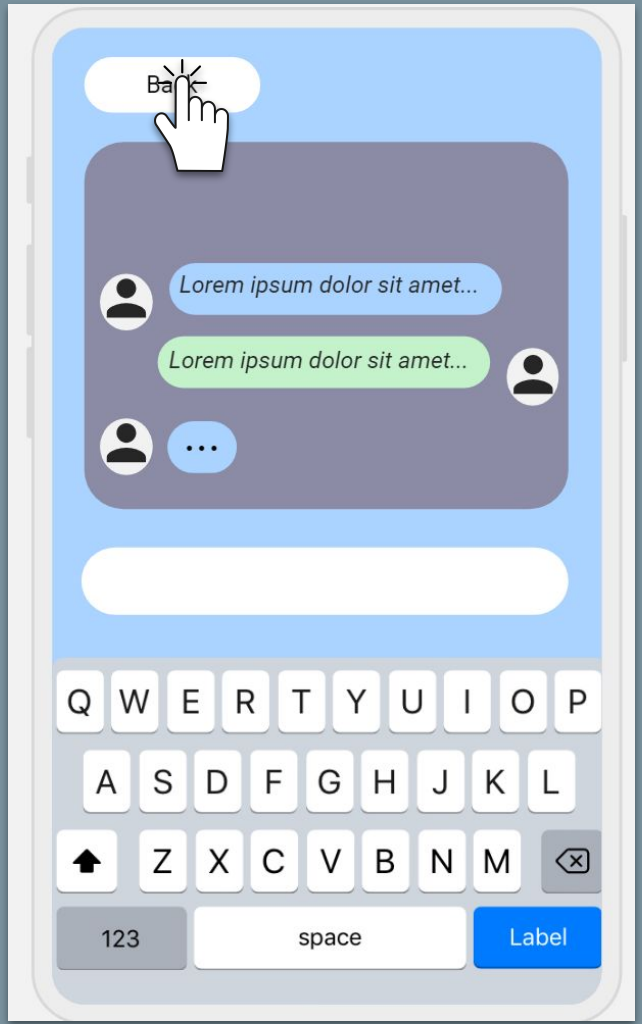

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
Cancel
Save

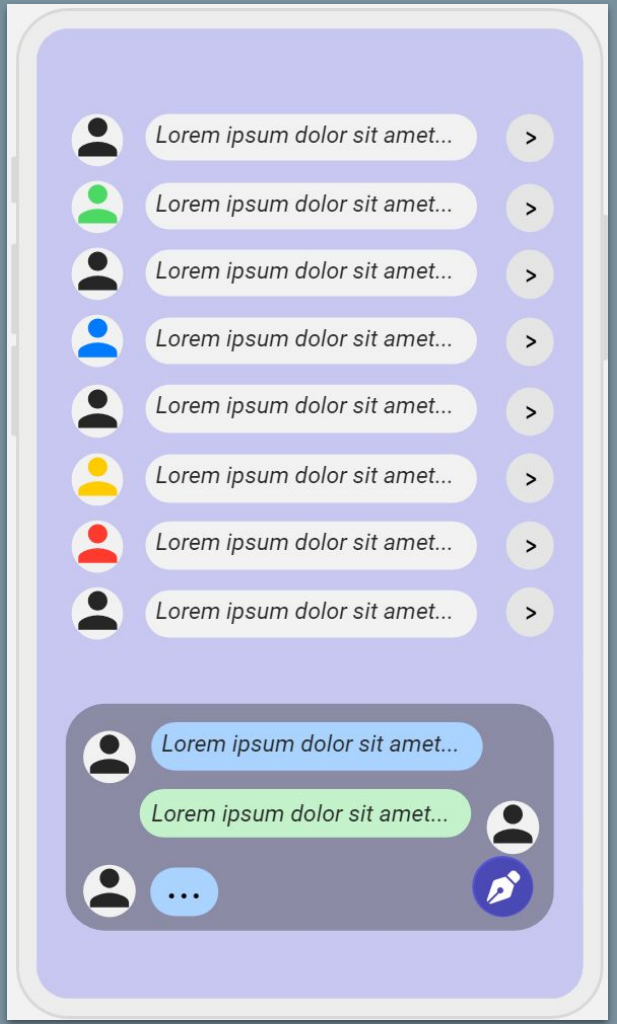

Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...


Back
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
def generate_fibonacci(n):
fibonacci_sequence = [0, 1]
while len(fibonacci_sequence) < n:
next_number =
fibonacci_sequence[-1] +
fibonacci_sequence[-2] ...
View code
Edit task


def generate_fibonacci(n):
fibonacci_sequence = [0, 1]
while len(fibonacci_sequence) < n:
next_number = fibonacci_sequence[-1] +
fibonacci_sequence[-2]
fibonacci_sequence.append(next_number)
return fibonacci_sequence
def main():
n = 20
# Number of Fibonacci numbers to generate
fibonacci_numbers = generate_fibonacci(n)
print(f"The first {n} Fibonacci numbers are:)
for number in fibonacci_numbers:
print(number, end=" ") if __name__ ==
"__main__": main()
Back
Attach code

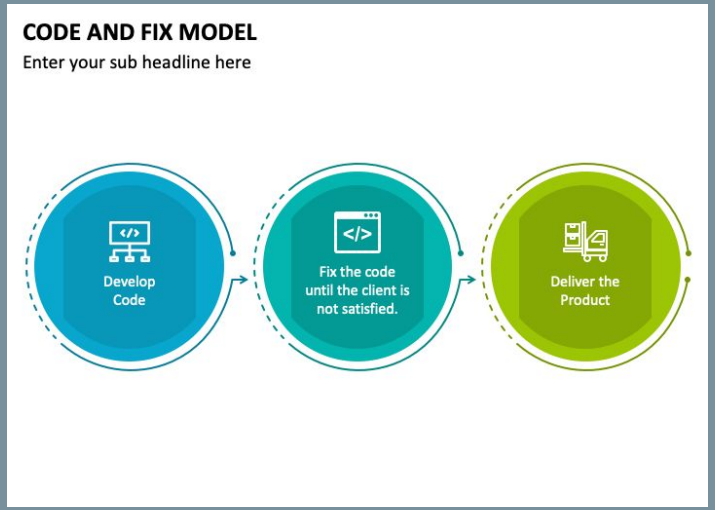

Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
...


Back
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
...
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M
123
space
Label


Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
Lorem ipsum dolor sit amet...
...

Processes and Tools Used

- Code and fix method
- Scrum meetings for planning work
- Pipe-and-Filter architecture pattern for program structure



Key Insights

- How to collaborate in a group setting with specific constraints
- Code and Fix is not the best approach with working with a larger group of people
- Opportunities for constant growth and improvement through Code and Fix

Limitations and Future Work

Limitations

- System status is sometimes unclear
  - What was the most recent edit?
  - Which task is the most recent?
- Certain UI elements have inconsistent formatting
- Version control safeguards are not implemented

Future work

- Connecting to code repositories
- Live code updates
- User hierarchy
- Task history
- Version control

Related Works

- Workvivo
- Slack
- Asana