

Keep

積極的な発表

議論を活発に行えた

できることをやったこと

実行委員に相談して、すぐに解決することができた。

口頭で出した意見を簡易的な議事録のようなものにまとめた

時間管理が出来て、振り返りの時間をしっかり取れた

コミュニケーションを取りながら実装を進めることができた

コミュニケーションをとって、試行錯誤しつつ戦略を考えることができた

困りごとは早めにチームで共有できた

こまめにチームで現状の共有をする。

考えこむよりも手動かして実装ができた

スプリントごとに目標に沿って、動くものができた

Try

場を盛り上げ

チームのメンバーとの認識が合っているかが常に確認できるように、合意事項はメンバーが共有できるところにまとめておく。

画面共有などのツールを積極的に使って認識の同調率を高める。

自分の担当部分だけでテストできるようにする。(他の人の作業待ちのような時間を作らない)

進捗に貢献する

2
1
1
1
1

Problem

タスクの工数見積もりが甘かった

タスクの達成条件があいまいでタスク+aに入ってしまった

開発に集中して、周りの進捗状況を把握できなかった

時間にルーズだったこと

オンラインでのコミュニケーションの難しさ

全体の進捗を気にするばかりになってしまい、自分の報告が出来なかった

一人の作業に集中してしまい、周りのフォローができなかった

戦略をもとに実装してみると予測できない課題が発生して多くの時間を要した

方向性を決めるのに悩んで、実装への着手が遅れた。

単純な質問でのアラートが多かった。

話し合いの際に他人の意図の理解や自分の意図が伝わりづらかった

他のタスクとの線引きや達成条件を把握する。

方針を固めて、うまくタスク分けをする

時間配分に気を付ける

チームだけでなく、実行委員にも質問する

T: ふりかえりごとに工数の見積もりをつけなおすとかタスクの分割を検討する

残り時間を考えたうえで、できること/しなければならぬことを選択してタスクに取り組む

Keep

チームのミッションを達成できた

アイデアを出し合いながら楽しく開発や戦略立てをすることができた

モデル比較で、見える化を意識して、結果を計測することができた

一度、全員で戦略を練る時間を設けるべきだった。

走行テストなどの作業をできるだけ分担してできた

役割が無い人も自分で見つけたりしたおかげで、役割分担を明確に素早く行えた

Try!

問題をすくに共有したい

わからなくなったら、わかっている人に聞く

スプリント開始にスクラムマスター、タイムキーパを意識的に決めておく。区切りの時間を明確にしておく。

スクラム開発を行うときには役割をはっきりさせて行う

人見知りせず話せたこと

コミュニケーションがとれていた。(質問・指摘しやすい雰囲気)

オンラインながらチームメンバーと密に連絡が取れた

課題や悩み事の共有ができるようになった

常に不明点や疑問点は質問をするようにして、何とか最終的に動くものを実装できた。

必要な情報やルールはあらかじめ共有する、ないものは定義する。

タスクごとの作業時間の見積もりをする。

スクラムの工程と時間管理に意識を向ける。

掛かる工数について見通しの修正をデイリースクラム等で行う。

Problem

すべての調整がうまくいったわけではなかったため、最後は運任せになったこと

最終的にどんな動作をするのか、全体的なテストが行えなかった

自分のステートマシン図が見つからいせいで、内容の説明が難しかった。

実装に集中しすぎてしまい、時間管理が疎かになってしまった

相手が話しやすい雰囲気を作る

自分で後から見返したり、他人に伝えるときにわかりやすいものを常に書くようにする。

とりあえず動くものを作ることに集中して、あとからモデルを見返したときにわかりづらいモデルになってしまった。

最後のスライドで図を使えば戦略がもっと伝わりやすかったと思う

スクラムマスターがあまりできなかったこと

分割して作る際に、状態名や変数名のつけ方を事前に統一していればスムーズに結合できた。

新しいタスクに進んで着手することができなかった

実装している時に、時間管理やタスク管理がおざりになる

3月14日