Project Proposal

Project Title : TraWell

Guided By : Professor Khushru Doctor

Group number: 10

Jeet Shah 006, Kesha Bagadia 011, Manav Vagrecha 022

Manal Shah 026, Parth Patel 028, Yashvi Gandhi 033

Shreyansh Shah 046, Mihir Chauhan 065, Yashvi Pipaliya 092

Topic: E-commerce website for community car-rental service

With the cities expanding and endless roadlines, transportation is becoming an issue for the people who do not have their own personal transport and in this information age, where anything and everything is facilitated through online means, we plan to build a Online Car Rental Website. If someone needs a car for a short period of time, they can find one closest to their choice and range on here. Even if one's on a vacation in an unfamiliar place, this site will provide a trusted and convenient source car rental. And if someone has a car lying around that they don't use, they could profit off it by lending it. So this software will be for the users who would like to book cars as well as for users who'd like to lend cars, all within a few clicks. The navigation and the purpose of the website will be simple and accessible across multiple devices. The users will also be able to look up for the cars they need by company, price etc.

Features

- A user can Lend a spare car on our portal.
- The user will be provided with a diverse choice for renting the cars.
- Users can filter cars by categories like vintage, sports, sedan, suv, etc.
- There will be a user account to rent or lend a car.
- For renting a car, users will have to create an account or login to their existing account.
- Filtering / Sorting Cars using bifurcation parameters like car type, automatic / non-automatic, cars details: company, colour, price range, mileage, capacity, model.
- To ensure safety, a renter can only rent a single car at a specific time.
- Personal Information of the Renter will directly be sent to the Lender as soon as the renter requests for the Booking.

Stakeholders

- External Stakeholders
  - Car Landing Customers
  - Other online car-rental services
    - AVIS India
    - Zoom Car
    - Ola Rental
    - Drivezy
    - Revv
    - Uber
  - Rentee Customers
  - Car Companies
  - Investors
  - Gas / Fuel Stations
  - Car repair and maintenance services

- Internal Stakeholders
  - Company Managers / Board of Directors
  - Developers / Developer Team
  - Designers / Designer Team
  - Content Manager
  - Insurance Company
  - Professor Khushru Doctor
  - Anupama Nair
  - Muskan Matwani

Technology Stack :

MERN Stack

Selected Model : **XP (Xtreme Programming)**

Extreme programming (XP) is a Software Development Methodology which is intended to improve software quality and responsiveness to changing customer requirements. As a type of agile software development, it advocates frequent “releases” in short development cycles, which is intended to improve productivity and introduce checkpoints at which new customer requirements can be adopted. We have chosen this because it has an array of advantages that fit well to our cause.

The Reasons Behind Selecting XP:

- This model is suitable for groups of smaller size which fits appropriately for our functioning.
- This model is completely flexible to the changes in requirements.
- It's lightweight, efficient and supple. This helps us to follow the procedure with ease.
- Extreme programming also addresses the project risk by frequent and shorter development cycles and consequently enabling early feedback.
- Reviewing the code at every step in every iteration along with regression testing at every stage of development will make the process more effective.
- Daily reusing of codes will make the design more productive and short iterations will make delivery more effective.
- The customer will be actively involved with the team to perform continuous planning, testing, and feedback.
- Structured framework activity with various phases namely, Planning->Design->Coding->Test->Release
- Circular loop concept, so that phases can be repeated if required.
- A KISS (keep it simple, silly!) design phase, due to the use of CRC cards based on the Object-Oriented Approach
- The design phase encourages refactoring so that the external behaviour is not affected due to the changes and improvements in the internal structure
- Unit tests are developed prior to the code.
- Pair programming helps us to avoid frequent mistakes and find better test cases.

Why Agile Over Other Process Flows:

Waterfall

The Waterfall model is the earliest SDLC approach that was used for software development. The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete.

Disadvantages:

- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are at a moderate to high risk of changing. So, risk and uncertainty is high with this process model.
- It is difficult to measure progress within stages.
- Cannot accommodate changing requirements.
- Adjusting scope during the life cycle can end a project.
- Integration is done as a "big-bang. at the very end, which doesn't allow identifying any technological or business bottleneck or challenges early.

Evolutionary - Spiral

It is a combination of iterative development process model and sequential linear development model i.e. the waterfall model with a very high emphasis on risk analysis. It allows incremental releases of the product or incremental refinement through each iteration around the spiral.

Disadvantages:

- It is not suitable for small projects as it is expensive.
- It is much more complex than other SDLC models. Process is complex.
- Too much dependable on Risk Analysis and requires highly specific expertise.
- Difficulty in time management. As the number of phases is unknown at the start of the project, so time estimation is very difficult.
- Spiral may go on indefinitely.
- End of the project may not be known early.
- It is not suitable for low risk projects.
- May be hard to define objective, verifiable milestones. Large numbers of intermediate stages require excessive documentation.

Evolutionary - Preprocessing

Evolutionary process model resembles the iterative enhancement model. The same phases are defined for the waterfall model occurring here in a cyclical fashion. This model differs from the iterative enhancement model in the sense that this does not require a useful product at the end of each cycle. In evolutionary development, requirements are implemented by category rather than by priority.

Disadvantages:

- Not suitable for smaller projects.
- Management complexity is more. End of project may not be known which is a risk. Can be costly to use.
- Highly skilled resources are required for the risk analysis. Project's progress is highly dependent on the risk analysis phase.

Incremental

Incremental Model is a process of software development where requirements are broken down into multiple standalone modules of software development cycle. Incremental development is done in steps from analysis design, implementation, testing/verification, maintenance.

Disadvantages:

- Needs good planning and design.
- Needs a clear and complete definition of the whole system before it can be broken down and built incrementally.
- Total cost is higher than the waterfall model.

XP Over Other Agile Models:

Scrum

Scrum is a process framework used to manage product development and other knowledge work. Scrum is empirical in that it provides a means for teams to establish a hypothesis of how they think something works, try it out, reflect on the experience, and make the appropriate adjustments. That is, when the framework is used properly. Scrum is structured in a way that allows teams to incorporate practices from other frameworks where they make sense for the team's context.

Disadvantages:

- Scrum often leads to scope creep, due to the lack of a definite end-date
- The chances of project failure are high if individuals aren't very committed or cooperative
- Adopting the Scrum framework in large teams is challenging
- The framework can be successful only with experienced team members
- Daily meetings sometimes frustrate team members
- If any team member leaves in the middle of a project, it can have a huge negative impact on the project
- Quality is hard to implement until the team goes through an aggressive testing process

Kanban

Kanban is a visual system for managing work as it moves through a process. Kanban visualizes both the process (the workflow) and the actual work passing through that process. The goal of Kanban is to identify potential bottlenecks in your process and fix them so work can flow through it cost-effectively at an optimal speed or throughput.

Disadvantages:

- Keeping the Kanbans resized as demand changes can be slow and difficult to manage.
- Lost hardcopy tags/forms and Kanban cards. As these are integral to the Kanban system loss or damage to them may result in process difficulties. To compensate for this a variety of electronic software based Kanban systems are available. Ideally to mitigate this Kanban cards should be kept on in-going and outgoing racks.
- Utilising the Kanban system results in less opportunity to work ahead of schedule placing performance pressures on staff.
- It can be inflexible to product mix changes and small or infrequent orders do not fit the model well. It could be argued that the Kanban system is more suitable for medium to large sized organizations.

- In order to facilitate the logistic process of quickly moving material to numerous workstations on the production line, a clean and well organized environment is required. A failure to do this may degrade the Kanban process.

Scrumban

Scrumban is an Agile development methodology that is a hybrid of Scrum and Kanban. Scrumban emerged to meet the needs of teams who wanted to minimize the batching of work and adopt a pull-based system. A hybrid of Scrum and Kanban gives teams the flexibility to adapt to stakeholder and production needs without feeling overburdened by their project methodology. Scrumban provides the structure of Scrum with the flexibility and visualization of Kanban, making it a highly versatile approach to workflow management.

Disadvantages:

- As Scrumban is a relatively new concept, there are no best practices for using it. This can result in it being used differently across teams with no standardisation.
- In Scrumban teams choose which task they work on, and so it can be difficult to track individual team member's contribution and efforts.
- Although the Project Manager has long-term control over the process, once the tasks have been identified and prioritised it is the project team who handle and implement them. As there are no daily scrum meetings the Project Manager doesn't get an overview of progress as it happens.

Agile SAFe:

SAFe is most-popular among enterprise organizations as many of its facets focus on eliminating the common challenges teams face when scaling agile.If one's established company is just beginning to transition to agile, SAFe might be a viable option to bridge that gap because of its more prescriptive approach

Disadvantages:

- SAFe is particularly popular with larger enterprises.
- It takes too much of a top-down approach which removes front-line players from the decision-making process and even from one another. This distance can limit their understanding of the entire software development lifecycle and hinder their ability to conduct well-informed and collaborative planning sessions as a result.
- This model requires too much upfront planning and process definition.

Agile LeSS

Large-Scale Scrum (LeSS) isn't a new and improved Scrum. And it's not Scrum at the bottom for each team, and something different layered on top. Rather, it's about figuring out how to apply the principles, purpose, elements, and elegance of Scrum in a large-scale context, as simply as possible. Like Scrum and other truly agile frameworks, LeSS is "barely sufficient methodology" for high-impact reasons.

Disadvantages

- LeSS can be used for large scale enterprise.
- It requires at least 8 teams with 8 members in each team.

Agile DSDM

DSDM is an agile project framework made up of 8 principles, a life cycle, products, roles and responsibilities and several best practice techniques. These underpin and support a philosophy of delivering strategically aligned business benefits as early as possible to give an organisation the best possible return on investment

Disadvantages

- Involves progressive development of requirements
- Focus on RAD can lead to decrease in code robustness
- Requires full commitment to DSDM process
- Requires significant user involvement
- Requires a skilled development team in both the business and technical areas
- Probably the most heavyweight project compared in this survey.
- Expects continuous user involvement
- Defines several artifacts and work products for each phase of the project; heavier documentation.
- Access to material is controlled by a consortium and fees may be charged just to access the reference material.

UP

Unified Process is based on the enlargement and refinement of a system through multiple iterations, with cyclic feedback and adaptation. The system is developed incrementally over time, iteration by iteration, and thus this approach is also known as iterative and incremental software development.

Disadvantages:

- The team members need to be expert in their field to develop a software under this methodology.
- The development process is too complex and disorganized.
- On cutting edge projects which utilise new technology, the reuse of components will not be possible. Hence the time saving one could have made will be impossible to fulfill.
- Integration throughout the process of software development, in theory sounds a good thing. But on particularly big projects with multiple development streams it will only add to the confusion and cause more issues during the stages of testing.