**Feasibility Report**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Revision #** | **Date** | **Description** | **Author** | **Approver** | **Comments** |
| **1** | **23/3/2013** | **This is the first version of the feasibility report.** | **Team**  **Phoenix** | M.V.Padmasri |  |

1. **INTRODUCTION**
   1. **Overview**

The product, Event4u is a web application. Creating the product would take a specific amount

Of time, effort and technical skills. This document provides an almost accurate picture into the amount

Of effort and time taken to build the project thus deciding its feasibility.

* 1. **Project Scope**
     1. **The Problem**

Non-availability of a software that would assist a common man in successfully organising an event.

* + 1. **Scope**

A web based event manager that would simplify the whole process of organising an event by helping the organizer divide the co-organizers or volunteers into different teams, allocate different tasks to them, supervise and coordinate between them. The views would be different depending on the roles of the teams and our software would have a feature for the event manager to track the expenditure. A portal for the participants of the events to put across their queries to the event managers would also be included.

* + 1. **High level system requirements**

**Event4u** must be delivered within 27 April 2013

* + 1. **Constraints**

**Event4u** must be delivered within 27 April 2013. The server for the deployment of the product has not been purchased as on date 27/3/2013.

* + 1. **Team coordinator/Interface for the project**

Navaneeth Y.V

* + 1. **Sponsor/People Involved with the project (other than the students)**

None

* 1. **Documentation/Literature Survey Summary**
* https://docs.djangoproject.com/en/1.5/
* http://www.djangobook.com/en/2.0/index.html
* http://docs.python.org/2/tutorial/
  1. **References**
     + **www.extremeprogramming.org/rules.html**
     + **www.extremeprogramming.org**
     + **SRS v2.0**
     + **User stories**

1. **Project Approach**

Event4u will be built using Extreme Programming based on the scenarios gathered during requirement analysis. The approach is chosen based on the deadlines given for the project completion and the modular structure of the project. Event4u will have 3 important work divisions in implementation, namely the model (Database part), the view (The Control logic) and the Template (Presentation). The above described structure allows the division of the team in a way so as to support paired programming. Small releases, pair programming and sustainable pace made Extreme Programming a better approach for Event4u.

1. **Potential Solutions**
   1. **Required resources**

The software, Event4u works on Mac, Windows and Linux and is platform-independent. The languages used for the implementation of the product, Event4u are HTML, CSS, JavaScript and Python.

* 1. **Cost/benefit analysis**

|  |  |
| --- | --- |
| Characteristics | FP count |
| Backup and recovery | 1 |
| Data communications | 5 |
| Distributed processing | 0 |
| Performance critical | 2 |
| Existing operating Environment | 1 |
| Online data entry | 4 |
| Input transaction over multiple screens | 3 |
| Master files updated online | 4 |
| Information domain values complex | 3 |
| Internal processing complex | 2 |
| Code design for reuse | 3 |
| Conversion/Installation in design | 1 |
| Multiple installations | 0 |
| Application design for change | 4 |

DI=Degree of Influence

Hence, Technical Complexity Factor (TCF)=0.65+0.01DI = 0.65 + 0.01 \* 33 = 0.98

**Unadjusted Function Points (UFP)**

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Simple | Average | Complex |
| Input(I) | 3 | 4 | 6 |
| Output(O) | 4 | 5 | 7 |
| Inquiry(I) | 3 | 4 | 6 |
| Logical Internal(L) | 7 | 10 | 15 |
| Interfaces(I) | 5 | 7 | 10 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Information  Domain Value | Pessimistic  Count | Most  Likely  Count | Optimistic  Count | Estimate  Count | Weighting factor | FP  Count |
| Number  of  external inputs(I) | 5 | 3 | 2 | 3 | 4 | 12 |
| Number  of  external outputs(O) | 6 | 4 | 3 | 4 | 5 | 20 |
| Number  of  external inquiries(E) | 7 | 5 | 3 | 5 | 4 | 20 |
| Number of internal logic files(L) | 4 | 3 | 2 | 3 | 10 | 30 |
| Number of  external interfaces (F) | 4 | 2 | 1 | 2 | 7 | 14 |
|  |  |  |  |  |  | 96 |

UFP = 4I + 5O + 4E + 10L + 7F

UFP=96

Thus, Function Point (FP) = UFP \* TCF = 96 \* 0.98 = 94.08 .

cost per person per month = Rs 25,000.

Fp per month = 6 fp/pm.

Man months taken to complete the project = 94.08/6 = 15.68 man months.

Cost taken to complete the project = 25000 \* 15.68 = Rs 3,92,000.

Man months/person = 15.68 / 7 = 2.24 man months / person.

**Expected duration**

The product, Event4u is expected to be completed by April 27,2013.

* 1. **Major deliverables**

Since extreme programming is being used, design is done based on iterations. The first iteration is scheduled on 26/03/2013 after which 1st set of design document will be produced.

* 1. **Project Plan Deliverables**
     1. **Feasibility Report**

Event4uwill be online event management software that makes the event managers task effective and manageable. The software provides event creation options and good co-ordination between the teams. The cost involved in handling the event can also be handled effectively. The above mentioned features will be delivered through extreme programming according to the schedule described.

Technical Feasibility

In the implementation of Event4u there are 3 important work divisions, namely the model (Database part), the view (The Control logic) and the Template (Presentation). The above described structure allows the division of the team in a way so as to support paired programming. The project structure is most suited to be implemented by a framework, rather than from scratch.

1. Since django uses the MTV, that is the Model Template View format; Django will be a better choice for the application development.
2. There are frameworks such as Ruby on Rails which work on similar lines. But since the team is familiar with Python rather than Ruby, Django is the most well suited option.
3. Django is built on python. Python is a high level, well documented language which is easier to learn. The choice of python facilitates better technical feasibility as the team is familiar with the language.
4. The web development will involve usage of CSS, HTML and JavaScript. The whole team is learning all these technologies as a separate course. This will increase the technical feasibility of the project to a great extent.
5. Small releases, pair programming and sustainable pace made Extreme Programming a better approach for Event4u.
6. Event4u as a web application is more feasible than as a desktop application. The application will be deployed on a local server of the firm. This makes the product maintenance easier.
7. As a web application there is a chance for application of web analytics to gather user data. This gives a better picture of user requirements needed for project improvement in future.

Time feasibility

Depending on the cost estimation, the time span required for the project is 2.24 man months per person. The deadline for the project was announced as 27 April, 2013.

1. The schedule has been designed by keeping the above mentioned points in mind.
2. Since it is a team of 7 with 2 testers, it was decided that the project Event4u can be implemented within the deadline.
3. Django, the technology to be used is well documented and has an easy learning curve. This would facilitate us to complete the project, Event4u within the deadline.
4. The division of work among the team would bring about parallelism in the project. This makes the project more time feasible.
   * 1. **Requirements Document**

User stories and SRS are attached with this document.

* + 1. **Design Document**

Since extreme programming is being used, design is done based on iterations. The first iteration is scheduled on 26/03/2013 after which 1st set of design document will be produced.

* 1. **Use cases for the solution**

User stories are attached with this document.

The user story to be implemented is specified after iteration planning scheduled on 26/3/2013.

* 1. **High level risks and challenges**

**Risks**

* Change of python and django version during the course of the project.
* Unavailability of a server for deployment.

**Challenges**

* Effective learning of the new tool, that is django.
* Finishing the product with in the deadline.
  1. **Time commitments from Team**

The team is committed to work on the product, Event4u till April 27,2013.

* 1. **High level schedule with Milestones**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sl.No | Phase | Activity | Start Date | Planned End Date | Who all | Comments |
| 1. | Requirement analysis | Collect data from end users. Release SRS v1.0 | 28/1/2013 | 31/1/2013 | Complete Team Phoenix | To interact with event managers to gather the necessary requirements |
|  | Requirement analysis | Define the problem statement and understand the user requirements | 31/1/2013 | 5/2/2013 | The whole Team Phoenix | Spend more time on understanding the need of the user and to freeze the requirements. |
| 2. | Planning | Decide on technical and time feasibility,  work breakdown, | 6/2/2013 | 8/2/2013 | Project Manager | To review various process models. |
|  |  | Decide on project schedule and a process model. | 8/2/2013 | 12/2/2013 | Project Manager | To work on  team dynamics and work division |
| 3. | Designing  and Coding | Envision the UI by manually drawing UI. | 12/2/2013 | 27/2/2013 | Designers | To get a clear picture of how the UI would look like. |
|  | Designing  and Coding | Implement a basic working model of the UI. | 12/2/2013 | 27/2/2013 | Front end Developers  as a pair. | To have a basic webpage designed for an event manager and a todo list for him.  (The subsequent week won’t be available for the firm because of predefined commitments) |
| 4. | Designing  and Coding | Have a model of each actor in every use cases described in SRS.  Build a database  for Event4u that models all the actors. | 12/3/2013 | 21/3/2013 | Designers,  Backend developers as a pair.  Front developers as a pair | Build a budget tracker.  Learn to use django and to have the application program interface ready from the backend. |
| 5. | Coding and  testing | Integrate front end and backend of the design.  Build the test cases to evaluate the product. | 2 21 /3/2013 | 1/4/2013 | Front end  Developers and the backend developers.  Pair of  testers | (Depends on 4 and 5)  Have  a complete  package of  working  Model. |
|  | Coding and  testing | Based on the  test results make  necessary  changes to  the code. | 1/4/2013 | 4/3/2013 | All the  implemento  rs and  testers. | Bug fixing  period.(  The subsequent week won’t be available for the firm because of predefined commitments) |
| 7. | Testing, Software  refinement and  product  release. | Make  the product  add any user  requested  features(if any). | 15/4/2013 | 27/4/2013 | Complete  team  phoenix | Bug fixing  and final  review period |

* 1. **Interaction and Communication plan within and outside the project**

The team is constantly engaging itself with interactions with people outside to get better requirements details. The detail of such interaction is mentioned in SRS. Such an interaction is an add on to our extreme programming approach as requirements keep on changing. The team is having meetings/ discussions either offline or online. Meetings in essence will build the team better. As a part of extreme programming the implementation is always done in pairs. The above mentioned feature makes the communication between team members essential and effective.

* 1. **Assumptions**

The user is familiar with internet and usage of web applications.

* 1. **Other considerations**

**NA**