Case-B 项目文档-User1

组别：工具组

项目文档

# Introduction

# 第一章：引言  
  
## 背景  
  
在当今城市化快速发展、交通运输模式日益复杂的背景下，高效、便捷的城市出行导航工具成为社会各个阶层的迫切需求。City Mapper应运而生，旨在通过智能化的交通导航解决方案，为用户提供实时准确的路线规划、公共交通时刻表查询、交通状况分析及兴趣点探索等功能。旨在打通交通出行信息流，从而让城市出行变得更加高效、便捷、环保。  
  
City Mapper不仅提供基本的出行导航功能，还通过强大的分析能力和深度的城市交通数据集成，提供个性化的出行建议和优化路线。City Mapper的用户群体广泛，包括有至少需要高效便捷的城市通行、常常需要查询交通状况并做出最佳路线选择的市民、学生、商务人士及旅游者。City Mapper在提升城市通勤效率的同时，还有助于降低交通拥堵带来的环境问题，提倡绿色出行，为构建智慧城市提供了强有力支持。  
  
从市场竞争角度来看，City Mapper致力于打造一个集成了多元交通方式、实时数据更新、智能化优化路线规划的品牌，从而区别于竞争对手，如谷歌地图、苹果地图、百度地图等，突出了City Mapper的特色和优势，为用户提供了更加个性化的交通出行服务。这不仅包括对地铁、公交线路、共享出行工具（如共享单车、共享汽车）和步行路线的全面支持，还包括了对实时交通状况的高级分析和预测，以便用户在规划行程时能够做出最佳选择。City Mapper通过提供这样的特性，不仅满足了城市居民的出行需求，而且促进了更加高效、环保的城市交通系统发展。  
  
## 业务目标  
  
本文档旨在明确City Mapper的软件需求，为开发团队提供明确的指导和方向。通过提炼用户需求、明确设计目标以及优化功能实现，City Mapper将继续优化用户体验，针对交通导航领域持续提升，不断推出更高效、更智能、更人性化的产品。通过整合多种交通方式的信息，并为用户提供实时的导航和规划建议，City Mapper将在长期内致力于成为城市出行的首选导航应用，提高城市居民的出行满意度。  
  
## 范围  
  
本文档覆盖City Mapper所有功能需求、非功能需求以及相关的约束条件，旨在为系统的开发、测试和部署提供全面的参考。并且，City Mapper的功能需求分别涵盖了：起点和目的地的创建与管理、路线规划、交通方式选择、距离查询及计算旅行时间、用户偏好设置、实时交通状况更新和查询时刻表。此外，文档还涵盖了管理员账户的创建与管理以及兴趣点的探索功能。软件还遵循相关的行业标准，并注意到用户隐私保护。  
  
## 定义、缩写与缩略语  
  
为确保交流与理解的准确性，对于文中使用的各类专业术语、缩写和缩略语，下表给出了其明确的定义。以此作为文档中词汇含义的依据，避免可能产生的歧义。  
  
| 缩写/缩略语 | 定义 |  
| ------------ | ------------------------------------------------ |  
| UI | 用户界面（User Interface） |  
| API | 应用编程接口（Application Programming Interface） |  
| GPS | 全球卫星定位系统（Global Positioning System） |  
| DB | 数据库（Database） |  
| REST | 表示状态转移（Representational State Transfer） |  
| HTTP | 超文本传输协议（HyperText Transfer Protocol） |  
| SSL | 安全套接字层（Secure Sockets Layer） |  
| POI | 兴趣点（Point of Interest） |  
| ODBC | 开放数据库连接（Open Database Connectivity） |  
| CRUD | 创建（Create）、读取（Retrieve）、更新（Update）、删除（Delete） |  
| RSS | 真实简单聚合（Rich Site Summary） |  
| DBMS | 数据库管理系统（Database Management System） |  
| SQL | 结构化查询语言（Structured Query Language） |  
| HTTPS | 安全超文本传输协议（HTTP over SSL） |  
| MIME | 多用途互联网邮件扩展（Multipurpose Internet Mail Extensions） |  
| XML | 可扩展标记语言（Exstensible Markup Language） |  
| HTML | 超文本标记语言（HyperText Markup Language） |  
| JSON | JavaScript 对象表示法（JavaScript Object Notation）|  
| URI | 统一资源标识符（Uniform Resource Identifier） |  
| URL | 统一资源定位符（Uniform Resource Locator） |  
  
## 参考标准  
  
City Mapper的设计和开发过程严格遵照了下列标准和文档，确保了开发方向的正确性以及系统功能和性能的基本要求。  
  
1. Internet Engineering Task Force (IETF) 标准文档  
2. 英特网组织 (IETF) : 使用HTTP/HTTPS等网络协议支持安全高效的数据传输。  
3. W3C Web技术和标准: 采用Web标准(如HTML5, CSS3, JavaScript)保证可访问性和兼容性。  
4. Open Geospatial Consortium (OGC) 标准 : 遵从GIS相关标准，保证地图和位置服务的全面集成。  
5. 数据库设计三大范式: DBMS设计依照第1、2、3范式保证快速、安全的数据操作。  
6. ISO/IEC 25051:2016:评估软件使用质量标准。  
7. ISO/IEC/IEEE 12207:2017:系统及软件工程 — 生命周期规程。  
8. GB/T 16260.1-2016:软件工程 — 产品质量要求与评价。此标准由中国国家标准化管理委员会发布，针对软件工程的质量进行了详细规定和要求，涵盖质量特性、度量方法、评估过程等多个方面。  
9. ISO 9001:2015: 质量管理体系要求标准。确保产品开发和交付过程具有一致性和可靠性。  
10. CN ASIL (中国功能安全等级) ：在处理安全关键功能（如实时路况更新）时，确保符合相关功能安全标准。  
11. 显示屏幕使用时间和健康标准：如IEEE P1789标准，用以评估和减轻长时间使用City Mapper过程中可能带来的健康风险，保证用户在使用软件时的健康。  
  
通过对这些文档和标准的引用和遵循，City Mapper不仅能够提供高质量的软件服务，而且能够确保用户数据的安全性和隐私性，以及系统的稳定性和兼容性。

# Overall Description

### 第二章 总体概述  
  
#### 2.1 产品视角  
  
City Mapper是一款革命性的智能出行规划软件，旨在通过综合化、智能化的服务，满足用户多样化的出行需求。其愿景是构建一个智能化、个性化的综合交通出行平台，让用户在出行过程中享受到更为便捷、精准、智能的服务体验。目标用户群体包括但不限于经常需要跨城市、区县出行的日常通勤者、旅游者、学生以及对外出自习从而避免堵车现象的短途出行者。City Mapper期望在市场中占有领先地位，成为人们日常生活中的出行助手与规划专家，通过技术创新解决出行的痛点，从而在日益拥挤与复杂的都市交通环境下脱颖而出。  
  
#### 2.2 产品功能  
  
City Mapper的核心功能模块包括但不限于起/终点管理和创建、路线规划、交通方式选择、距离查询、计算旅行时间、用户偏好设置以及实时交通状况更新。其中：  
  
- 起/终点管理和创建允许用户轻松添加、管理和修改出行起点及终点，显著提升用户使用过程中的灵活性和便利性。  
- 路线规划功能能够综合分析距离、交通状况及时刻表信息，为用户推荐最适合其出行需求的路线。实时更新交通状况和动态调整计划中的公路状况或公共交通状况，确保用户信息的准确性。  
- 交通方式选择模块通过结合用户的偏好及其他出行要素，提供多种交通方式建议。用户可以根据需要选择公共交通、自驾、步行或是骑行等多种出行方式。  
- 距离查询和旅行时间的计算为用户提供直观的地理信息视图，特别在设定行程规划场景时，可以辅助用户有效地做决策。  
- 用户偏好设置支持个性化的交通方式偏好设置，体现了City Mapper对于用户体验的关注度和致力于个性化服务的决心。  
- 实时交通状况更新确保了用户在出行前获取到最新最准确的交通信息，增强出行的预见性和安全感。  
  
在这基础上，软件还提供了时刻表查询功能供用户及时了解公共交通的运行时间及状况，探索兴趣点功能鼓励用户多角度发现城市文化景观。这些功能相互交织、相互补充，共同构成了City Mapper强大功能集合，力求全面提升交通出行服务的质量。此外，还包括兴趣点的探索与管理，可以用户按需查找相关点，主动触发系统展示信息，进一步提高用户的体验度。  
  
#### 2.3 用户特征  
  
City Mapper的主要用户群可以分为两大部分：普通用户与管理员用户。作为普通用户，他们能够轻松地位于系统管理的起始点和目的地，同步设置了敏感因子，如通勤时段的交通流量及城市景点在周末的访问量。这将影响系统如何推荐最佳路线或交通工具。广泛具有一定的互联网操作设备条件，具备较强的自我管理能力，不畏惧盲从，愿积极发掘新鲜移动APP。此外，他们也有一定的个性化偏好，包括对于自身旅行时间或公共交通线路的兴趣等，这些都需要被考虑在内。对于管理员用户而言，具备专业的技术操作与管理的权限，主要关注数据更新、运营维护等功能以对起始/终点、兴趣点等数据进行有效的管理和综合展示，注重系统稳定性与连续性，愿意花费时间为整体表象付出努力。  
  
#### 2.4 竞品分析  
  
市面上已有诸如Google Maps、百度地图、高德地图等众多成熟的出行规划类应用。这些竞品多关注于提供高质量的路径规划和实时交通状况更新，但经常会缺乏对用户偏好的深度理解。相比这些成熟的产品，City Mapper通过深度零星结合技术、大数据与人工智能技术来理解和优化用户的个性化偏好（如趋向公交与拥堵程度），还可以探索城市的兴趣点和深入景点介绍，以期提供一种更个性化的服务。面临较强竞争压力，这将使产品面临提高精度、响应速度等技术挑战，但这也将进一步推动功能的深化，提供更为优化的体验，进一步提升用户粘性。  
  
#### 2.5 业务风险与机遇  
  
从业务风险来看，尽管City Mapper当前所依托的技术在同领域内相比处领先地位，但仍面临数据处理考验，如何在保持高精度和响应速度的同时保护个人隐私和其他数据，确保系统的稳定性都可能会成为一个问题。此外，在客户或政府机构短期内对技术的认知度较低的情况下推动其业务发展的难度相对较高，本质上会影响在市场上获得竞争优势。  
  
机遇方面，随着大数据及AI技术不断发展，再加上城镇化发展模式以及人口密度增加所带来的交通压力，城市出行管理与规划将越来越受到重视。这将是智能运载系统增长的关键时期，技术创新压力将会成为City Mapper未来持续增长的关键因素。而其持续性将成为City Mapper的核心竞争力，在今天及明年的推广中占据竞争优势。  
  
#### 2.6 设计约束  
  
技术栈要求是基于多种编程语言和框架的混合开发模式，以利项目迭代与系统性能优化。支持主流操作系统与硬件平台如iOS、Android设备，有助于拓展更大范围的用户群体。法规方面，必须确保收集、存储及处理的数据严格按照《个人信息保护法》与《数据安全法》的要求执行，以确保用户隐私安全。  
  
#### 2.7 假设与依赖关系  
  
业务开展前期假定本公司当前所具备的技术实力与项目管理能力可以确保项目按时按质完成。开发过程中需要与其他团队如数据收集与处理团队、数据库维护团队密切协作，确保数据来源的准确性与及时性。此外理论上尽可能减少政府权限上的阻力，更易推进这些创新技术被社会接纳。作为第三方服务组件，Vendor需确保数据库查询等外围工具的功能正常使用，满足软件运行要求，以最小化系统整体构建与运行中的变动与风险。

# Functional Requirement

- 用户注册  
 - 描述：新用户可以通过输入基本信息来创建账户。  
 - 输入：用户的姓名、电子邮件地址、密码。  
 - 输出：新创建的用户账户，用户可登录到系统。  
   
- 用户登录  
 - 描述：已注册的用户通过输入姓名和密码验证身份后登录系统。  
 - 输入：有效的用户名及其关联密码。  
 - 输出：登录到系统。

-创建起始点  
 描述：用户可创建新的起始点坐标和地址信息。  
 输入：起始点的详细信息（如地址、名称）。  
 输出：数据库存储用户创建的起始信息。  
  
-管理起始点  
 描述：管理员可更新或删除系统中的起始点。  
 输入：起始点的标识符、需要更新的信息或删除的操作。  
 输出：系统中的起始点信息更新或被删除。  
  
-创建目的地  
 描述：用户可创建新的目的地坐标和地址信息。  
 输入：目的地的详细信息（如地址、名称）。  
 输出：数据库存储用户创建的目的地信息。  
  
-管理目的地  
 描述：管理员可更新或删除系统中的目的地。  
 输入：目的地的标识符、需要更新的信息或删除的操作。  
 输出：系统中的目的地信息更新或被删除。  
  
-规划路线  
 描述：用户可以根据起点和目的地信息获取由City Mapper应用推荐的路线列表。  
 输入：起点和目的地的详细信息。  
 输出：系统提供的起点到目的地的推荐路径。  
  
-选择交通方式  
 描述：用户可以选择特定的交通方式参与路线规划。  
 输入：用户的选择（如公共交通、步行、骑行等）。  
 输出：系统存储用户的选择，并作为进一步分析的基础。  
  
-距离查询  
 描述：用户可以查询起始点和目的地之间的直线或道路距离。  
 输入：起始点和目的地的详细信息。  
 输出：两地之间的距离信息，以千米或英里表示。  
  
-计算旅行时间  
 描述：用户可以计算从起始点到目的地的预计旅行时间，基于选择的交通方式。  
 输入：起始点、目的地、和交通方式的详细信息。  
 输出：预计的旅行时间。  
  
-设置用户偏好  
 描述：用户可以设置其在交通方式和路线上的偏好。  
 输入：用户偏好信息（如偏好使用公共交通而非单车）。  
 输出：存储于数据库中的用户偏好设置。  
  
-实时更新交通状况  
 描述：系统将实时更新交通状况，并反映在应用中。  
 输入：来自外部的实时交通数据源。  
 输出：数据库中最新的交通状况信息。  
  
-查询时刻表  
 描述：用户可以查询相关交通方式的时刻表信息。  
 输入：查询条件，例如特定交通方式、站点名称和日期等。  
 输出：符合查询条件的时刻表信息。  
  
-探索兴趣点  
 描述：用户可以浏览或搜索各种兴趣点，如景点、餐馆等。  
 输入：兴趣点的查询条件。  
 输出：系统提供的搜索结果，包括兴趣点的名称、地址和类别等信息。  
  
-创建和管理管理员账户  
 描述：为系统管理员创建账户并进行账户管理（更新或删除操作）。  
 输入：管理员账户的详细信息，更新或删除操作。  
 输出：系统中管理员账户信息的更新或删除。  
  
-用户偏好记录  
 描述：系统记录并持续更新用户偏好设置。  
 输入：用户的偏好选择。  
 输出：存储在数据库中的用户偏好记录。  
  
-实时交通状况记录  
 描述：系统存储和更新实时交通状况数据。  
 输入：交通状况的数据源信息。  
 输出：数据库中最新的实时交通状况记录。

# External Description

### 第四章 外部接口  
  
#### 4.1 用户接口（UI）  
  
本节概述软件与最终用户之间的接口。用户通过图形用户界面（UI）与软件交互。City Mapper的设计必须满足跨平台兼容性，提供一致且优化的用户体验。  
  
- 操作界面：包括但不限于创建和管理起始点及目的地、选择交通方式、查询距离及旅行时间、设置用户偏好以及探索兴趣点等功能的部分。  
- 界面响应性：UI设计需要确保快速响应用户的交互，支持多种输入方法，如触摸、点击和手势。  
- 适应性设计：确保UI在不同设备中和操作系统中的一致性显示，并且支持可定制的视图选项来优化不同用户的需求。  
- 数据展示：将实时交通状况、导航路线规划和其他重要信息以用户友好且直观的形式展现，提供信息图表、地图和其他可视化工具。  
  
#### 4.2 系统接口（API）  
  
City Mapper与外部系统（例如，地图API、交通实时数据API）交互，以提供导航和交通信息服务。这些接口是实现系统核心功能的关键。  
  
- 地图服务API：为路线规划和展示用户当前位置提供地图视图。  
 - 通信协议： RESTful API通过HTTP/HTTPS进行通信。  
 - 数据格式： JSON用于层端数据传输。  
 - 使用案例：规划最优出行路线。  
- 实时交通信息API：获取起点和目的地附近的实时交通状况，帮助用户进行路径优化。  
 - 通信协议： RESTful API通过HTTP/HTTPS进行通信。  
 - 数据格式： JSON。  
 - 使用案例：实时更新地图上交通拥堵的路段信息。  
- 数据库接口：系统使用数据库来存储和检索用户数据、偏好设置、交通时刻表等信息。  
 - 数据库类型：利用关系型数据库(SQL)和/或NoSQL数据库（例如MongoDB或Cassandra）根据具体需求进行了适当的选择。  
 - 访问协议：应用SQL语句或者采用NoSQL数据库特有的API进行数据查询、修改操作。  
 - 数据格式： SQL语句，以及NoSQL适用数据模型。  
  
#### 4.3 数据接口  
  
City Mapper不仅依赖内部数据库存储数据，还需要外部实时数据源获取动态信息，比如交通实时状况的更新，此类外部数据源头包括但不限于城市交通管理中心或者其他第三方通讯公司。  
  
- 数据交换格式：优先选择开源标准和广泛采用的数据交换格式，比如XML, JSON, CSV等进行数据交换以增强兼容性。  
- 获取数据：通过API接口（例如REST，WebSocket等）来实现实时信息更新，确保数据的实时性和准确性。  
- 数据更新频率：实时数据的更新频率依赖于数据源以及系统的特定需求。一般而言，针对实时交通数据，更新频率可达到每分钟甚至更短。  
  
#### 4.4 安全接口  
  
系统对外接口需具备高水平的安全性，确保用户信息不被泄露及数据传输的安全性。  
  
- 安全机制：借助SSL加密技术保护用户数据传输的隐私。  
- 标准遵循：严格遵守ISO/IEC 27001信息安全管理体系标准，进行风险评估来检查及减少任何潜在的安全漏洞。  
- 用户权限管理：确保只有授权人员才能访问某些系统功能，比如管理起始点或目的地、记录用户偏好等。  
  
#### 4.5 可访问性接口  
  
保证软件产品可以为所有用户提供无障碍访问，无论用户的物理、感官或认知障碍。  
  
- 文本到语音：对于视觉有障碍的用户，提供文本转语音的支持。  
- 语音控制：允许用户通过语音输入互动，便于操作软件界面，例如通过Siri, Google Assistant等。  
- 放大选项：图形和其他UI元素必须能够被放大或缩小，以满足有不同视觉需求的用户。  
  
综上所述，本章概述了City Mapper与不同外部系统间的交互接口，旨在通过规范保证用户、系统与外部源之间高效可靠的数据交互，以提供卓越出行服务的同时确保安全隐私并提高硬件兼容性和用户体验。

# Non-Functional Requirement

### 第五章 非功能需求  
  
#### 5.1 性能需求  
  
City Mapper旨在提供高效、准确且实时的出行导航服务，满足不同用户群体的各种使用需求。以下是City Mapper的主要性能需求。  
  
1. 响应时间  
  
 - 定义：响应时间是指用户在系统中发起一个请求到收到系统返回的响应所消耗的时间。  
 - 衡量方式：在开发、测试阶段，使用专用的性能测试工具（如Apache JMeter、LoadRunner）对系统进行压力测试和性能基准测试，记录并分析各个功能模块的响应时间。  
 - 目标：  
 - 基础功能（创建起始点、管理目的地等）：在95%可服务请求时限中不超过1秒；  
 - 复杂功能（路线规划、交通状况同步等）：在95%可服务请求时限中不超过3秒。  
 - 高峰时段（如早、晚高峰）：  
 - 系统需在用户集中使用期间能够保持至少80%的响应时间不超过3秒，95%的响应时间不超过5秒。  
  
2. 系统吞吐量  
  
 - 定义：吞吐量是指单位时间内系统能够处理的请求数量。  
 - 衡量方式：通过压力测试工具对平台的并发用户数进行逐步递增测试，确定系统并发用户数与吞吐量指标的线性关系。  
 - 目标：  
 - 在单日最高请求数量峰值100万次的情况下，系统要能够维持不低于98%的处理请求成功比率。  
 - 高峰时段：  
 - 系统应当在早晚高峰期间具备承载不低于50万的日处理请求数的能力，并以不低于90%成功率完成请求处理。  
  
3. 并发用户数  
  
 - 定义：并发用户指系统同一时间内可以处理的用户请求数量。  
 - 衡量方式：使用性能测试工具模拟大量的用户同时登录和并发使用的场景进行测试，关注请求时延和错误率。  
 - 目标：  
 - 在并发2万用户使用情况下，仍能够保持每个功能模块不低于95%的成功率。  
 - 高峰时段：  
 - 系统在早晚高峰期间并发用户数为2万时的最大处理请求时间和错误率不超过10%。  
  
#### 5.2 质量属性  
  
1. 可靠性  
  
 - 定义：可靠性是指系统在特定环境下能够持续满足用户功能需求的能力。  
 - 衡量方式：设置实时监控系统的状态，并且对服务进行定期测试和维护，建立服务水平协议（SLA），以确保服务的一致性。  
 - 具体标准与目标：  
 - 每月平均硬件故障率不超过0.01%。  
 - 服务可用性达到99.99%，确保在任何10000分钟的运营时间内至少有9999分钟的时间可以正常访问City Mapper的服务。  
 - 对于系统部分组件出现故障，要求在其有意外中断4小时内恢复正常运行，最大中断时间不超过15分钟。  
  
2. 可用性  
  
 - 定义：可用性指的是系统降低不可用时间，确保用户始终能随时进行访问的服务特性。  
 - 衡量方式：记录非计划停机时间，关注客户投诉率，利用SLA建立服务水平协议。  
 - 具体标准与目标：  
 - 实行备份和容灾机制，确保当主节点系统发生故障的情况下，备份节点能够在5分钟之内接管其数据及请求流量。  
 - 系统能够在99.9%的时间内可被访问，对于无法正常登录0.1%，不可用性和维护总时间应不超过3小时。  
  
3. 可维护性  
  
 - 定义：可维护性是指系统能够便捷地进行测试、检查、改进和升级的程度。  
 - 衡量方式：计算软件维护成本，以及维护过程中遇到的错误量与严重程度。  
 - 具体标准与目标：  
 - 系统更新时尽可能确保用户操作层面的无缝过渡和兼容性，不给用户带来额外的使用负担。  
 - 对于所有非计划的缺陷，统计及控制其发现及修复速度（例如发现后1天内，修复总错误不超过5个，维护总时间不过4小时）。  
 - 通过持续集成和持续部署（CI/CD）流程，提升系统调试和维护的效率；每周功能迭代更新的平均完整性指标不低于95%，即更新后运行中发现的严重缺陷数量不超过5个。  
  
4. 可移植性  
  
 - 定义：可移植性是指系统的软件部署在不同硬件平台和操作系统下容易实现接入和兼容的能力。  
 - 衡量方式：进行跨系统测试，包括使用不同品牌和配置的设备，操作系统版本（包括但不限于Android、iOS、Windows、macOS）对系统在部署与功能发挥影响的试验。  
 - 具体标准与目标：  
 - 确认在主流设备操作系统平台（Android和iOS）上产品能正常运作，并兼容未来3年内的主流版本。  
 - 设计时考虑到适应性需求增加并实现灵活配置，以满足未来更新和扩展；测试在不同硬件设备、操作系统下稳定运行，并确保功能与性能表现。  
  
#### 5.2.1 可扩展性  
  
 - 定义：可扩展性是指系统能够处理更大的负载或者更高体积的数据的能力。  
 - 衡量方式：在现有容量基础上，逐步增加运行任务量或存储需求，通过监控资源使用情况来判断系统承受新任务的性能。  
 - 标准与目标：  
 - 系统在高峰时段涵盖增加10%以上处理量的任务，资源使用与响应时间误差不超过2%。  
 - 随业务发展，系统需具备快速便捷接入新功能和新服务的能力，支持未来两年内业务需求至少增加50%（用户数目、访问人次）。  
  
#### 5.2.2 安全性  
  
 - 定义：安全性是指系统防止非法访问，确保用户数据和个人隐私安全的能力。  
 - 衡量方式：对程序进行渗透测试（包括协议漏洞测试、操作系统安全测试、缓冲区溢出测试等），进行实时监控观测实施安全策略效果。  
 - 标准与目标：  
 - 全面数据加密服务（其中包括传输过程加密以及数据库信息存储加密），确保数据在传输与存储方面不会遭受未授权的访问和泄露。  
 - 每季度由外部专业安全审计机构进行一次全面的渗透测试，对任何发现的问题在一星期内完成修补和报告。  
 - 日志记录与事件告警：实时监听并记录潜在的安全事件并发出告警以便管理员进行处理；符合《计算机信息系统安全保护等级划分准则》标准，达到网络安全等级保护的三级要求。  
 - 用户隐私保护：严格按照有关隐私保护法律法规的要求，采用多层安全防御措施保护用户信息，遵循有效的数据管理策略和隐私政策；数据只用于改善用户出行体验，禁止将数据转移给第三方用于非平台用途，除非经过客户明确同意。  
  
#### 5.2.3 兼容性  
  
 - 定义：兼容性指软件能与各种网络、硬件环境、数据库、第三方服务等良好的工作程度。  
 - 衡量方式：在设计和开发过程中进行跨平台及跨操作系统测试，采用成熟的接口标准（如RESTful API）进行数据交换，以期实现无缝对接和信息交互。  
 - 标准与目标：  
 - City Mapper将全面支持所有主流的移动操作系统平台，包括但不限于Android5.0以上版本，以及iOS9.0及更高版本。  
 - 与主流的移动应用SDK（如Google Maps API、第三方支付接口、地图引擎）进行兼容性与集成的验证；在兼容性测试中，确保City Mapper在多操作系统、各系列品牌手机上测试运行。  
 - 根据全球市场趋势，硬件版本、操作系统和SDK版本的更新及时调整与更新系统的需求，以确保能跟上最新型号的用户需求，提高兼容性和工具兼容性。  
   
#### 5.2.4 用户界面设计（UI）  
  
 - 定义：用户界面设计指用户操作系统的友好性和易用性，是否直观、简洁、美观并方便用户操作。  
 - 衡量方式：通过用户体验测试、可用性测试，收集并分析用户反馈，以改善和优化界面设计。  
 - 具体标准与目标：  
 - 界面风格遵循一致性和简洁性原则，确保各种功能具备良好的可发现性、易学性和易用性。  
 - 每次版本更新对于UI界面决策内容收集综合用户从多角度的反馈，进行评估和调整，持续优化用户体验。  
 - 适配触屏设备的轻敲、滑动等手势操作；用户界面预计要能够在95%的设备分辨率上保持良好显示，避免屏幕分辨率切换引起界面畸变。  
- 5.2.5 法规遵从性  
  
 - 定义：法规遵从性指产品设计和操作均要严格遵守现行有效的数据安全法规和隐私保护法律。  
 - 衡量方式：针对草案，城玛实况平台与相关法律监管机构进行实时交流，确保所有服务、业务和运营过程与法律法规一致。  
 - 标准与目标：  
 - 城玛实况平台需要遵循强大法规，包括遵守《网络安全法》、《数据安全法》等要求处理相关问题。同时全面配套实现用户请求隐私等功能，守护用户的基本数据隐私与安全。  
 - 定期和长期免受到任何法律纠纷的影响，维护用户的合法权益和数据隐私。  
- 5.2.6 合别人性化（Human Factors）  
  
 - 定义：指系统设计时充分考虑人类生理和心理特点，便于理解和操作。  
 - 衡量方式：以人为本的全面用户友好的测试，主要通过专业的用户体验团队与专家进行评估，以提高用户满意度。  
 - 具体标准与目标：  
 - 进行多次认知可行性的研究与人体工程学的设计测试，并根据结果和反馈持续优化。  
 - 对系统进行深度用户界面操作名词和指导指示优化，将术语标注和使用层次维持在普通市民水平，便于用户理解使用，同时具备直觉的指南功能。  
通过综合且细致地衡量以上各项非功能需求，City Mapper能够确保高质量的服务和体验，提供一个更加可靠、安全、可维护和用户友好的交通导航系统，为提升城市出行效率和个人出行体验尽一份力。